SCANNING BEHAVIOUR AND PATTERN RECOGNITION

JOHN ROWLAND LISHMAN

PhD
University of Edinburgh
1975

ABSTRACT

Two basic models of human pattern recognition have been advanced: feature analysis and hypothesis testing. These can only be discriminated by looking at behaviour before recognition. This is studied here by having the subject scan with a pen that writes only where the (invisible) picture is black. Although a computer simulation shows that it is possible and efficient, subjects scanning capital letters and simple shapes with this technique rarely use hypothesis testing.

Acknowledgements

This research was carried out in the Bionics Research Laboratory of the Department of Machine Intelligence and Perception, University of Edinburgh. I was supported for three years by a studentship from the Medical Research Council, and for a year by the Nuffield Foundation.

I am very grateful to my supervisor, Dr Jim Howe, for his guidance, advice and encouragement, without which I should not have been able to present this thesis. I should also like to thank Professor Richard Gregory for his initial help and support.

My thanks are due to Stephen Salter who was an invaluable source of technical expertise and William Edmondson, who drew the shapes used in the first experiment.

CONTENTS

Chapter 1

INTRODUCTION

How do we see things? Men have always speculated on this problem, and have solved many parts of it; physical optics, the optics of the eye, the chemistry of photoreception, the physiology of the eye and of the visual pathway are understood in outline. These processes explain how the object gives rise to patterns of light rays, and how these are transduced and transmitted to the brain, but does not explain the central mechanism of perception.

This central mechanism can be studied by investigating its behaviour - psychology, by studying the structure and function of its components - physiology, and by constructing analagous machines - artificial intelligence.

Some of the early ideas of how we see were complicated and unsatisfactory; they included suggestions that objects threw out small copies of themselves (Empedocles), or that particles were thrown out from the eyes, spraying the surrounding objects (Plato). Since then it has been accepted that light is reflected from objects, is collected and focussed by a lens on to the retina, and is there converted into neural excitation. Physics encompasses knowledge of light and lenses, ophthalmology studies the shape and optical properties of the eye and physiology studies the mechanisms of the retina. These systems are not fully explained, but the basic workings are known. This knowledge does not provide us with an adequate explanation of how we see, because the problem of pattern recognition remains unsolved.

The meaning of pattern recognition must be clarified. Pattern recognition is the conversion of a spatially-ordered image and stored, meaningfully-ordered representation to knowledge, or in machine terms, a statement about the world, Figure 1.1.

The knowledge that results from recognition of an object, eg, a tiger, is of the form "That is a tiger". To know this the perceiver must already know about tigers, that is, have an internal representation of tiger, a pattern. This pattern and an image of the tiger must be connected to give knowledge "That is a tiger". This connection is pattern recognition. A pattern also related to information about the learnt properties of the object it specifies, and these learnt properties can be called up when the pattern has been connected with the object perceived. Thus it is not the visual properties of a tiger that are frightening, nor the concept of a tiger in abstract, but the prediction about the future based on the knowledge of tigers in general and of the presence of this particular one.

Pattern recognition is thus the conversion from a spatially ordered image to a statement. Lenses, cameras and retinas convert a picture from one form of spatial order to another. The defining feature of a spatially ordered image is that it is the spatial relations of the elements which convey the information in it.

Consider the representations of the dog, "Fido". A pencil of light rays emanating from him, a photograph of him, a retinal image of him and a neural pattern excited by a retinal image of him are all spatial images, where the spatial relations of the

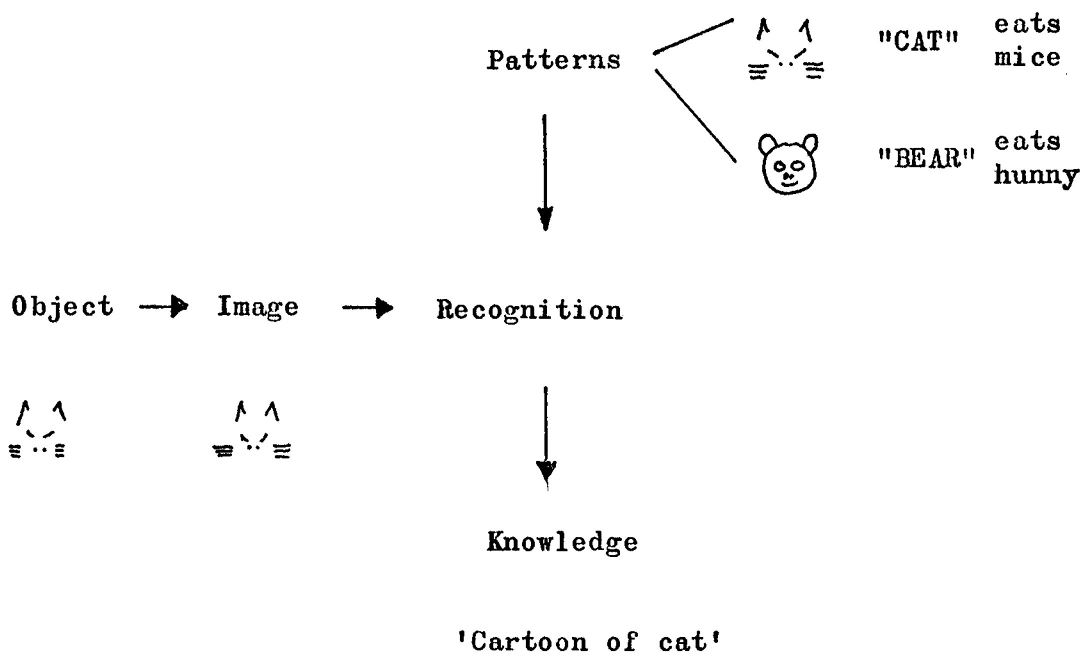


Figure 1.1

Object,image,pattern and statement.

images are preserved.

A pattern is a concept. Although this must have some physical embodiment, its relevant relations with other elements are ones of meaning, not of space. The pattern for Fido, his mental representation, relates to knowledge that he is a dog (class membership), a resident of my house (relation to observer), and likes liver (particular knowledge of that individual). This pattern for him must also include a representation of his appearance, although the form of this is unknown. Where, in a computer, in a brain or as words, these items are in spatial relation to each other, makes no difference to their import. They must, however, be linked properly to the master concept, "Fido", by, for instance, an index. My recognition system connects the image of the animal on the rug and the pattern for Fido, and thus gains access to learnt knowledge about him: I see Fido on the rug and know he will like the liver I brought him.

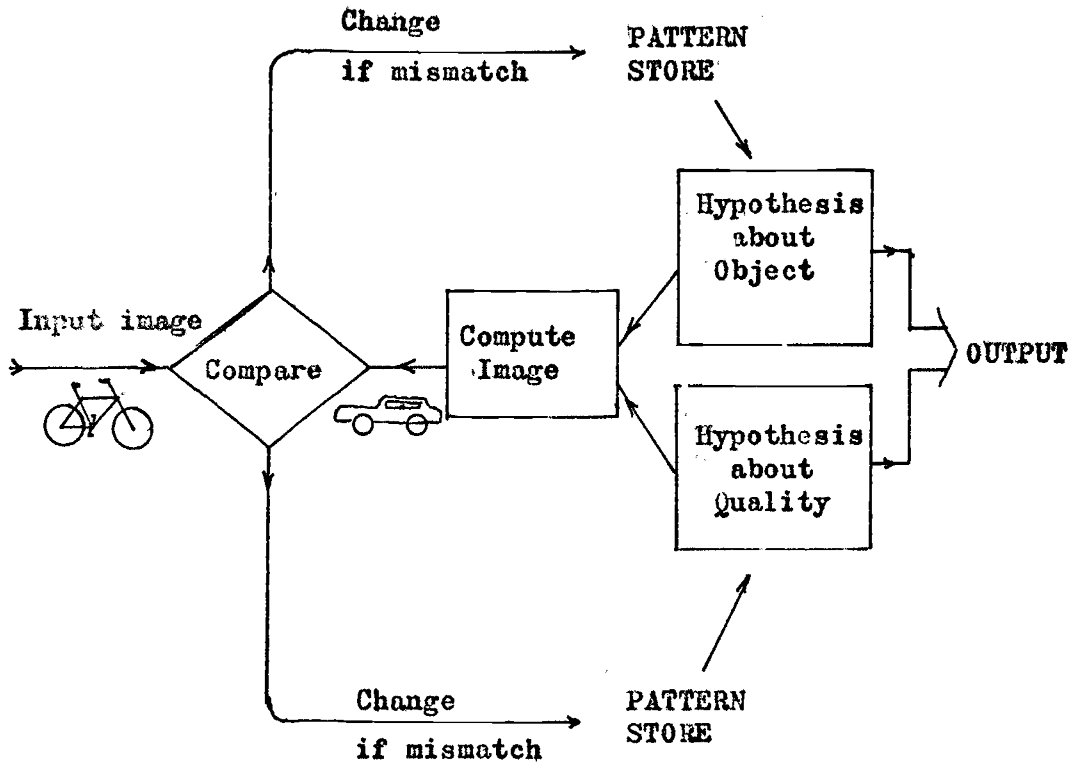
J J Gibson (1966) points out that many properties of objects are represented directly in the optic array. This accords with common observation; a large object approaching very fast is frightening, whatever it is. Its size, speed and time to contact are all potentially available information in the optic array (Lee, 1974). This, at first sight, appears to deny the utility of recognition.

However, some properties of the world, for example edibility, are not represented directly in the optic array, but must be learnt. To gain access to this stored information the object must be recognised. Also, the processes transforming invariants in the optic array to mental representations of the properties of the world are complex, and may usefully be regarded as recognition processes. The Gibsonian view may entail simpler pattern recognition processes, but does not eliminate the necessity for recognition as a conversion from a spatially ordered image to a meaningful statement about the world.

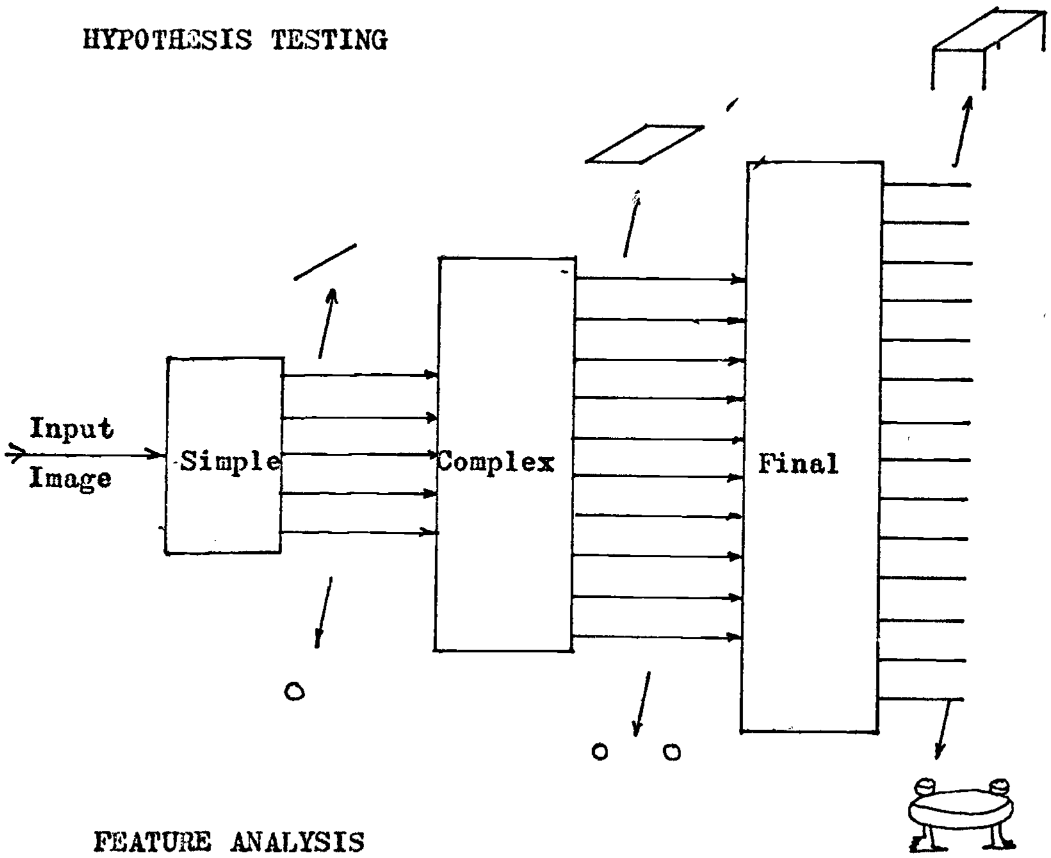
One reason why pattern recognition has not been successfully tackled in the past is that there existed no machines which could emulate the process. This implies that there was no expertise with such machines, and no language in which the solution could be couched, without an infinite recursion, where a homunculus looked at the input. The computer, hopefully, provides an adequate language and technology, to replace the homunculus with a machine whose workings are understood, but does not offer an immediate solution.

Gestalt psychology (Kohler, 1963) attempted to solve the problem of recognition, but suffered from lack of a machine to embody analagous processes. It emphasised wholes rather than parts, and suggested that images were represented as electric fields in the brain. This has been contradicted by the physiological evidence, but, more important, it is not a solution to the problem of recognition, because these fields are still spatially ordered. The translation to a meaningful statement has not been explained.

Two basic theories of pattern recognition have been advanced - feature analysis and hypothesis testing (Figure 1.2). The hypothesis testing theory argues that a hypothesis is made about the object - a pattern is selected, and that the pictorial representation in the pattern is checked against the image. If



HYPOTHESIS TESTING



FEATURE ANALYSIS

Figure 1.2

they match the object is recognised, and the pattern is the mental representation of the object. The source of the hypothesis is not specified, which is a major weakness of the theory, but can be based on prediction or the failure of previous hypotheses. It cannot be based on the properties of the image without another layer of recognition, because the image is spatially ordered and the hypothesis meaningfully ordered. This hypothesis testing model has been advanced by Richard Gregory (1966; 1970; 1973).

The basic proposal of feature analysis is that there exist feature detectors, which register the existence of some feature in the image (a feature might be a bright point, a line, a sharply curved object, a hand-shape or a person-shape). Higher level features are combinations of lower level ones. The difficulty with this approach is that it appears to require 'Mona Lisa' detectors, in many sizes, orientations and positions, ie, that a person can recognise the Mona Lisa implies, in this theory, a detector specialised in recognising this picture, and one for each of the many things, however rare, the person could recognise. A further difficulty is that the detector must recognise the object in a wide range of sizes, positions and orientations. This appears to imply that there is a multiplicity of detectors for each recognisable image, which could tax the capacity of even a human brain.

These two theories predict very similar states of mind after the image has been recognised. The behaviour of the system before final recognition is the only source of information able to differentiate between the two. The feature analytic system only responds to information presented to it, and some other mechanism must guide the search with no help from partial recognition. The hypothesis testing system searches for information in a manner dependent on what has been found; it will guide the search according to a current hypothesis which must be compatible with what is already known.

This study aims to find out whether the hypothesis testing model is used for recognition, and if so to investigate its operation. It will, therefore, require techniques designed to show that the subject bases his recognition tactics on what he has already found.

The philosophy underlying this study holds that two-dimensional static picture recognition for humans is a sub-category of all recognition, and is learnt as part of this more general ability. If the picture recognising mechanism is known, then recognition of the real world will be found to be an extension of this.

Objects in the real world move and impinge on all the senses, and patterns can be matched to information from all senses and their combinations, ie, heat, crackle and bright light equal fire. This study will concentrate on static visual two-dimensional images. This choice is one of convenience and interest, not necessity. Although this is a drastic abstraction from the real world, static pattern recognition is a normal ability of people, relevant to their management of everyday life. The Kodak company still makes profits on film to produce static pictures, which suggests that people find them useful. As such a human capability, static pattern recognition seems worth investigating in its own right.

If picture perception is a subsection of general recognition, then all people, even uninstructed ones such as primitive tribesmen or babies, should be able to recognise pictures. The evidence here is mixed.

Pictures and script tend to be artefacts of civilisation, and civilised people have little difficulty in recognising objects in pictures. They are rarely deceived into thinking that the picture is the real object: the rule-proving exception is trompe-l'oeil wall painting.

Deregowski et al (1972) found that members of a remote Ethiopian tribe had difficulty in recognising line drawings, and review similar reports from other remote populations. It could be argued that this shows that recognition of pictures is a learnt ability, analagous to the recognition of printed English, not a general ability. The evidence seems weak, complicated by other factors, such as the language difficulties of the tribesmen (did they have a word for picture?), the difficulties of translation (What question were they answering?) and the bad quality of the drawings.

Hochberg and Brooks (1962) raised their son with as little exposure to pictures as is possible in urban New England. He had no training in the meaning of pictures. At 19 months he was shown line drawings, then photographs of familiar things, and responded with the names of 13 or 17 (depending on the judge) out of 20 objects shown. This suggests that explicit instruction in the use of pictures is not necessary, and that picture recognition is a general ability.

This evidence is not enough to shake a belief that static pattern recognition is a subsection of general recognition, as well as worth studying in its own right.

Three approaches to pattern recognition are proceeding in parallel; artificial intelligence, physiology and psychology.

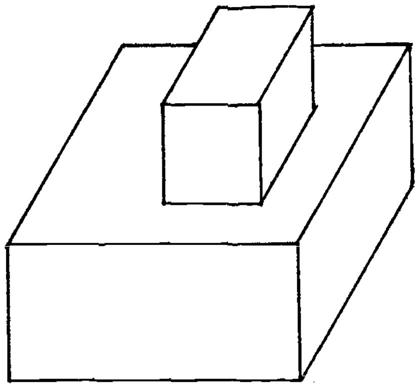
Artificial intelligence attempts to create pattern recognising machines of increasing power and generality. The tool is the computer, programmed to simulate each potential machine. This is certainly the way to test understanding of a proposed process; if it can be programmed then it is likely that the programmer understood the process.

The three problems which have received most attention are the analysis of line drawings, photograph interpretation and letter recognition.

Line drawings, particularly those with only straight lines, are easily represented in the machine, and pose an interesting, soluble but limited intellectual problem, that of working from a line drawing of stacks of blocks to the concept of the three dimensional objects, Figure 1.3.

Early programs to do this, such as Guzman (1968), worked ad hoc. Later work showed that the assumptions of this model corresponded to facts of projective geometry, and generalised them to more complex scenes (Winston, 1972). This model provided the universe in which Winston's (1970) program operated and demonstrated its capacity to manipulate relationships. It inferred from a number of examples that an arch is a block supported by two others with a hole underneath; this implies the creation and manipulation of hierarchical relational descriptions.

All of these attempts assumed perfect input, particularly that all relevant lines were there. Programs to produce these



is recognised as
BLOCK ABOVE BLOCK

Figure 1.3
Line drawing of the type recognised by Guzman's program.

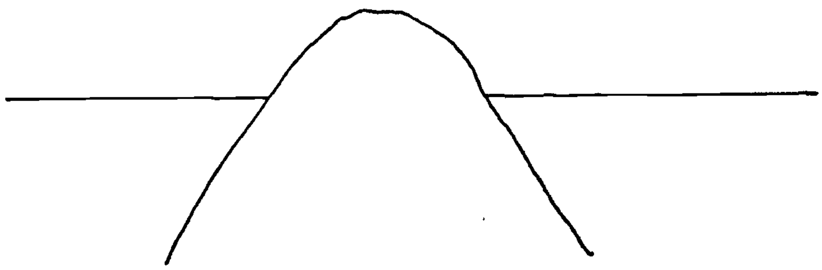


Figure 1.4
Picture of a grassy hump in a field. There is no boundary at the near edge because there is no texture change.

lines from TV pictures of real blocks, or to turn photographs into cartoons, have been unsuccessful (Sakai, Nagao, Fujibayashi, 1970; Murphy, 1969), but are now working adequately (Winston, 1972). The improvements in scene analysis and line finding involve a change in the program from a simple hierarchical feature analysing system to a heterarchical system with the ability to hypothesise. Kennedy (1974) working on the use of lines in cartoons or line drawings for human consumption, whether in instruction manuals or Punch cartoons, argues that lines represent discontinuities in the movement properties of the environment, as often as discontinuities in the surface texture or shadowing. These movement discontinuities are not available to most line finding programs, which means that they must use semantic information to differentiate a block edge from its shadow. The information in a cartoon is not a simple translation of the information in a picture, Figure 1.4.

Photograph interpretation is particularly motivated by defence demands. Consider the overworked Russian inspecting satellite photographs of the Los Angeles freeways for tanks, or his American counterpart inspecting square megametres of taiga. If basic analysis could be deputed to a machine, results would be produced faster, more accurately and with fewer personnel problems. Several contractors to the US Defense Department are working on this problem, but specialised techniques, such as holography, seem the most effective, and these are unhelpful to the understanding of human recognition.

In 1973 (Greenwood) the quality of satellite photographs was such that a tank and a truck were just differentiable by a human. The picture quality has improved since then. This illustrates conveniently one point in real applications, that one can sharpen either the receiver or the transmitter, and that improving the transmitter is usually easier. Thus the methods necessary to get better photographs from satellites, while complex and expensive, (diffraction limited lenses 3 feet in diameter!) are extensions of known techniques, better methods of interpretation are not. The point is clearer with letter recognition where one can either make the type to be recognised crisper, more standardised and better positioned, or make a recognition machine to accept imperfections. Commercial Optical Character Recognition equipment for computer input requires good print. One major potential user, the Post Office, cannot change the quality of the print it gets, and thus has had to consider better recognition techniques.

The normal technique of OCR is stencil matching, the simplest case of feature analysis. The stencil is represented electronically, and the comparison of this and a letter imaged on to a retina of photocells is very fast. This means that the system can cope with small mislocations of the letter by jiggling the stencil around, and still find that the limit on the speed of the whole is the rate at which paper can be moved. The system is not easily adaptable to non-standard typestyles or bad printing. (BCS, 1967)

A number of more advanced techniques are discussed and compared by Ullman and Kidd (1969). These are again designed to recognise characters on fast moving paper, and may be characterised as the seeking of sub-stencils (features) and their combinations. This enables the system to recognise moderately bad printing, but not different sizes and positions.

Systems for the recognition of handwritten characters are more complex. For separate characters and clean lines a system of feature detectors such as that of Uhr and Vossler (1961) is adequate. Characters, as written, even block capitals, are neither well separated nor composed of good lines, and more complex programs are necessary. For cursive script it appears that knowledge of the linguistic context is necessary to provide hypotheses which the machine can test against the script (Bornat & Brady, 1974). It should also be possible to adapt the machine to the style of writing being read. This has been found in other cases; Selfridge and Neisser (1963) found that Morse code could only be analysed by a detector with varying characteristics, because a dot in this word could be longer than a dash some minutes later, even in clearly readable Morse code.

In all these fields, mechanical pattern recognisers do not yet have the tolerance, ability to generalise or the reliability of humans. Machine recognition systems can be fast, efficient and very restricted, using feature analysis, or slow and more general using hypothesis testing. It is possible that the human combination of speed and generality is due to unknown, more sophisticated processes, such as hypothesis testing, rather than blockbusting by a myriad of cells working as detectors for all recognisable combinations of features.

The difficulty of creating artificial pattern recognition systems shows that pattern recognition in humans is not a trivial operation, even if people experience no difficulty in doing it.

The second major approach to pattern recognition is through physiology. Living systems do recognise patterns, and one can hope to find how it is done by observation of their internal workings. The technology needed for this is not yet developed. Neurons, the operational elements in the brain, are small cells with processes $\sim$ 10 m diameter. They can have 10,000 inputs and outputs. They exist in a tangled mass in a brain that pulses gently.

The normal tool for investigating these cells is the microelectrode, whose tip is about the same size as a nerve fibre. The procedure is to advance this into the nervous tissue of a restrained animal until it is within a cell, and then find what stimuli affect that cell. The power of the technique was demonstrated by Lettvin et al (1959), who showed that the frog's retina contained cells that responded to general illumination, large shadows (hawks?), small moving objects (bugs?). These correspond neatly to important aspects of the behaviour of frogs, who catch flying insects by flicking out their tongues, and have a response to predators of jumping to the darkest place visible.

Hubel and Wiesel (1962) found cells in the visual cortex of cats which responded maximally to lines, each cell having a preferred orientation and length. Other workers have extended these studies, and found cells responsive to movement, stereopsis and complex features (Blakemore, 1973). The line detecting cells appear to be a simple aggregate of on-centre, off-surround cells, themselves aggregates of basic photodetecting rods, and known to be widespread over the retina. This fits neatly with a hierarchical feature analysing model of pattern recognition, where recognition is achieved by noting the co-occurrence of features, which may themselves be co-occurrences of lower level features. This scheme is open to the objections noted above - Mona Lisa detectors and a combinatorial explosion. One pointer showing that this scheme is nevertheless a possibility is the monkey hand detector cell found by Gross et al (1972). No-one has shown a mechanism which would rotate, shift or magnify patterns, which is the machine builders method of reducing the number of detectors needed.

While the electrophysiological work neatly fits the requirements of a hierarchical feature analysing system, it does not specify it. As will be shown, the hypothesis testing model can be formulated in a way that tests the hypothesis, not against the raw input, but against a collection of deductions from it. Thus it would seem easier to test a sub-hypothesis (is there a line there?) against the output from a specialised detector (of lines), rather than against the output of many elementary detectors (of brightness).

A problem with the physiological approach is that it is much easier to find the details of the embodiment of a known operation than the workings of a system where both the function and the embodiment are not known. As a gedanken experiment consider analysing a machine from a sheet showing the connections between integrated circuits and the logical truth tables at the pins of each element. This is theoretically enough to specify the function of the machine, but is unlikely to be enough to understand it in practice. The problem is much worse if the truth tables are wrong or inadequate. It does become easier if it is known to be a disc controller, because some parts of its functioning are then known. This is a fair analogy to electrophysiology; the functions of a neuron are imperfectly known, and their connections are largely unknown. This interacts with a technological hazard, that current methods of investigation only allow a few electrodes at a time, and those not specifiably related. Thus it would be useful to look at the input and output of one neuron, but this facility is not available, except by accident. If information is conveyed inside the brain encoded as a pattern in parallel neurons, this could not be found from microelectrode studies.

The intrinsic limitations of the physiological approach, that it does not often differentiate between theories, and that deriving function from form is nearly impossible in practice, mean that a psychological approach, investigating the performance of intact animals, is very helpful before applying physiological techniques to show that a known mechanism has a determinable substrate.

In the psychological study of pattern recognition, feature analysis is the more obvious mechanism, and hypothesis-testing, a more complex theory, must be shown to fit the facts better if it is to be accepted. Two cases, in particular, are much more readily explicable by hypothesis testing than by feature analysis; cognitive contours and the hollow face. In Figure 1.5a feature analysis suggests one would see only what is there. However, subjects report a white triangle obscuring a figure, which must be a self-generated hypothetical construction. This triangle is matched with a brighter test patch than the white background of the same paper. A similar cognitive contour is powerful enough to show a Poggendorf illusion, Figure 1.5b (Gregory, 1973).

A hollow head, made like the inside of a mask so that the nose is a hole and not a protruberance, appears the normal convex

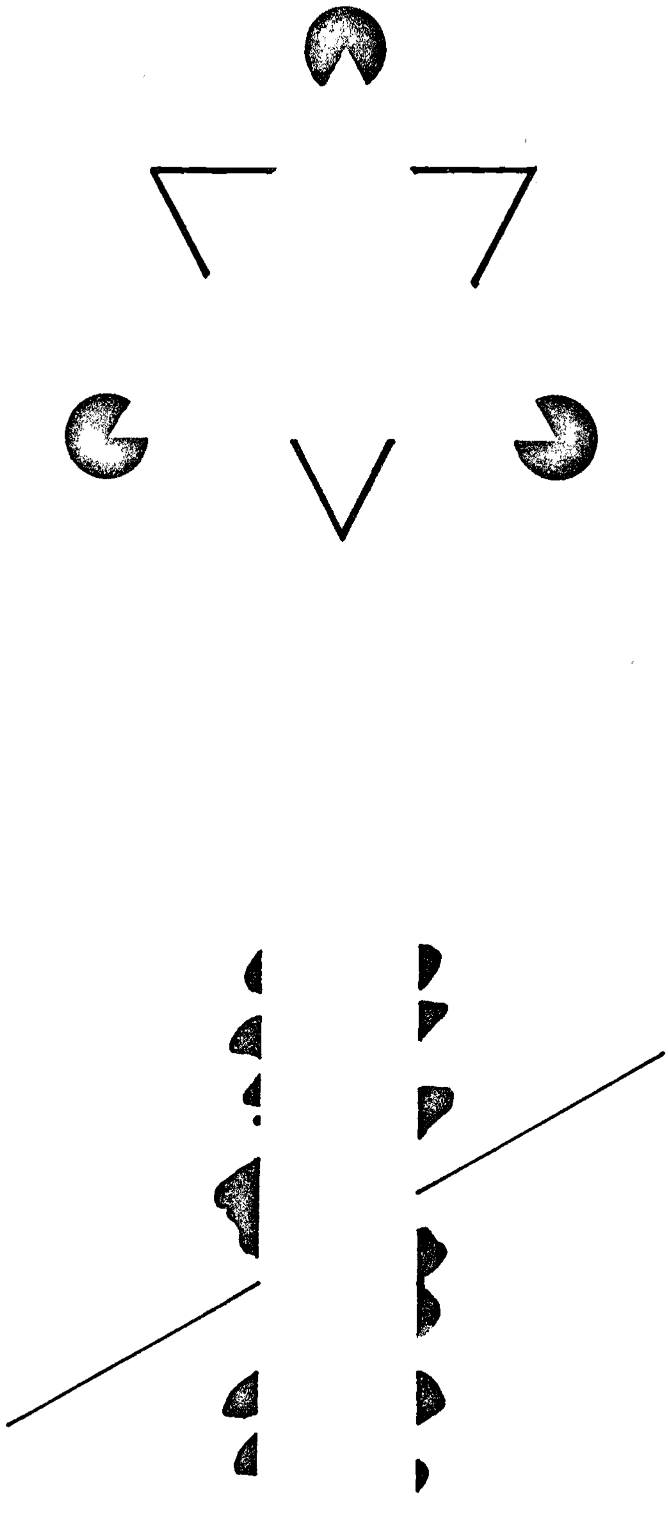


Figure 1.5
Cognitive contours

shape when seen from more than three feet away. The set of optical motions which would normally indicate depth by movement parallax are taken to indicate head turning. This is a powerful effect, and better explicable by hypothesis formation and testing.

To differentiate between feature analytic and hypothesis testing methods of pattern recognition, some response must be found before the subject recognises the object, because after recognition the predictions of the two systems are indistinguishable. Before looking at studies of scanning which are designed to answer this question, we will dispose of one potential confusion.

Many studies about scanning by humans are classified under "Visual Search". The term is used for a technique, that of timing a subject searching a field for a nominated item. The technique has been used to investigate a variety of problems of perception.

The time to find an item, as well as being a real practical problem in many fields, such as those of finding aircraft discussed by Morris and Horne (1960), can be processed to give information about the likelihood of any point being seen, given its prominence and position with respect to the fixation point, using as data the scan parameters.

In this way Bloomfield (1970) used the term "Visual Search" for the task of finding one target in a field with other elements, a task which introduces elements of pattern recognition. Bloomfield asked his subjects to find one different sized round dot among many, and from the search time deduced the area which was inspected at each fixation (of assumed duration). He found that as target dots and background dots approached each other in size, the area within which the difference could be detected at on fixation reduced. This is compatible with a retina on which those parts of the image far from the fovea are effectively out of focus. This could alternatively be a result of similar search areas at each fixation, but longer fixation times.

Another connotation of visual search is that used by Neisser (1967), who asked subjects to scan lists of letters, looking for target letters, or the absence of target letters, among others. From this investigation he showed that inspection time did not increase with the number of letters which could be targets, and thus ruled out sequential template (stencil) matching as a mechanism for letter recognition.

In these conditions the subject searches for the one different element. Bloomfield's subjects only had to remember what part of the field they had scanned; Neisser's subjects even had the order of search dictated by the form of the display, words on a page. The subject in these studies is using information gained so far to guide his scanning, but only so as to cover the field evenly. The visual search problem is best solved by a fast acting feature detector specialised only for targets. However, hypothesis testing can only manifest itself where the subject guides his scanning by information about an item which is not yet adequate for recognition of that item. In these visual search cases information about one particular element is either absent or adequate for recognition; that is, there is never partial recognition on which further information gathering might be based.

The most obvious way of looking at scanning and partial recognition is to study eye movements. Human eyes move to observe different parts of the visual field, because the foveal region of the retina, where acuity is best, is small, about 2 degrees across. The eyes are normally fixated, and move between these fixations in saccades or jumps. By looking at the pattern of eye movements one can hope to find what processes are operating to choose one place for the next fixation, rather than another. The problem has been expressed as "to look you have to see; and to see you have to look," meaning that you have to have some way of choosing the place to look without knowing what is there.

The possibilities for rules determining this behaviour are:

1. look where an elementary peripheral detector signals something of interest: this is most obvious where a motion seen peripherally elicits a saccade to fixate on it, but can happen with static elements such as sharp corners.
2. look somewhere determined by a general program. This would be a good strategy while searching a blank sky for an aircraft which could only be seen foveally. A suitable program would be to move in a regular space-filling scan.
3. look to where a feature of interest is expected on the basis

of what is already known - hypothesis testing. A simple variant of this suggests that one should move along lines of which part has been foveated. Others suggest that if a foot has been recognised there will be a head six foot lengths above it. If this process is driving eye movements it should be possible to work back from the scan to the concepts the subject had at each stage.

Eye movement studies agree that subjects concentrate on some parts of pictures, and scan sparsely over the rest. They concentrate on important parts of the picture, such as faces, which tend also to be parts with the richest detail (Yarbus, 1967). This is as predicted by a hypothesis testing formulation; there may be as many peripherally perceptible details in wallpaper as in a face, so peripheral driving would spread the fixations over both. Howe's (1965) work with random polyhedra as targets for eye movements studies shows that subjects appear to fixate preferentially on sharp corners, although this cannot be statistically validated. This is what would be expected if the subject fixated on peripherally perceived details. Howe also found that subjects scanning for a small low-contrast spot in an otherwise blank rectangular field used regular scan patterns, apparently perturbed versions of the well organised scans shown in Figure 1.6. Noton and Stark (1971) found that similar eye movement scans were used on different inspections of the same stimuli. They used large cartoons of low contrast, and the scan paths they publish vary greatly in detail. Peripherally driven eye movements would be expected to repeat the same path each time, hypothesis driven eye movements would be expected to follow

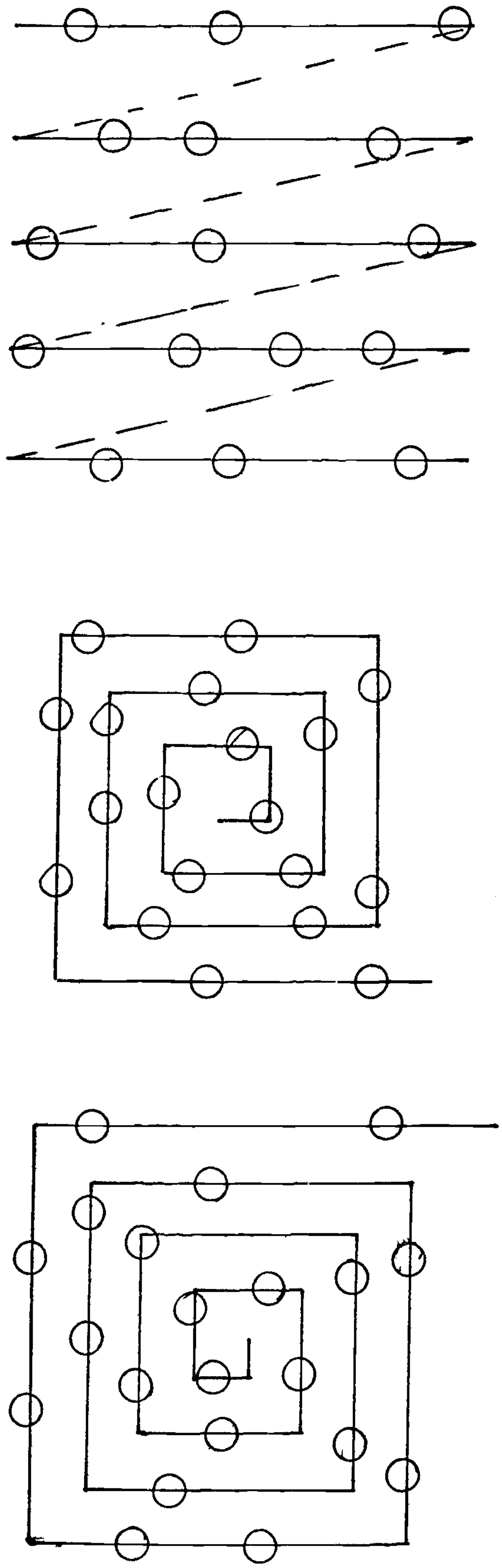


Figure 1.6

Idealised eye movement scans (after Howe) a path modified by the experience of the subject. The evidence is not good enough to separate these possibilities. These studies suggest that all three mechanisms are available to drive eye movements; the regular scan as a fall-back mechanism in a blank field, the peripherally-driven scan operating when the field is uninteresting, and the hypothesis driven scan when the image is familiar and interesting. These are suggestions, not proven.

The difficulty is in knowing how much has been taken in at each fixation, particularly from the periphery, and thus how much is already known. If the periphery of the retina is operative, it is virtually impossible to determine whether a given scanpath was driven by a peripheral mechanism or by hypothesis testing.

To complicate the interpretation of eye movements still further, Hackman and Guilford (1936) found a low correlation between eye movements and subjects reports of what was attended to. Mackworth (1965; Mackworth and Schissler, 1966) has shown that a subject may fixate a target without seeing it, implying that the information gained by a fixation is a function not only of the retina, but also of the subjects state of mind.

In view of our lack of knowledge of what is attended to and processed at each fixation, it seems unlikely that much information can be gained about the recognition system by eye movement studies. This does not indicate that eye movement patterns may not be useful as indicators of cognitive functioning in some circumstances. Tikhomirov and Posnyanskaya (1966) used eye movement patterns to elucidate the thought processes of chess players. They found movements which suggested that the player was looking at squares on the board to which he might move a piece, and at pieces which were in a position to attack that square. It is a popular cliche that "people's eyes give away their thoughts", and this suggests one mechanism by which they do so. Good chess players do not look at vulnerable positions.

These difficulties lead to attempts to use scans without peripheral operation which would only allow hypothesis-testing and regular scans to operate. This implies that the scanning element must be externally limited to exclude the periphery. It introduces the difficulty of artificiality into the situation, but makes clear the amount of information already exposed. Hochberg (1966) worked with the Judas Eye, a hole in a board which was moved in front of the pattern, revealing the pattern only through the hole. He used it to study perceptual memory, not scanning. The problem is that the picture available to the subject is only what is seen through the hole, now. He has to store the results from previous holes internally. We know that people store the results of eye movements to build up an internal picture of the world, but the ability to store the results of an external scan seems a dubious extension of this. This could be avoided by using eye movements to drive the Judas Eye, so that the 'hole' of the Judas Eye is always where the fovea is pointing. The technology of this is difficult.

A similar task is that of tracing outlines or solid objects with a finger. Blind people solve a task of this sort when they read raised lettering or maps by touch. The difficulties of this task are described by Leonard (1970). The task has also been investigated by Sokolov (1970), who found that a hexagon can be confused with a pentagon. This suggests that the mechanism of touch cannot readily differentiate angles of 108' from those of 120'. No start point is recognised on return to it, so a corner counting strategy is unworkable. The poverty of this mode of touch with full haptic perception (Gibson, 1966) is comparable with the relationship of the Judas Eye technique to normal vision, and as we have seen, it suffers the same problems.

The problem found with the Judas Eye is that the subject cannot remember what he has seen. This introduces one other form of scanning experiment, the one with which we shall be concerned. Here the scan leaves a trace, which can be inspected by the normal processes of sight. This means that the scan element can be well-determined, obviating any difficulties with peripheral perception. This enables one to determine whether the subject will use a programmed predictive scan, or will use one which varies with the part revealed so far, which would be characteristic of a hypothesis testing program. Because there is no peripheral information, the scan cannot be driven by it.

Two methods of doing this have been used; tile turning and moving pens. Tile turning (Gardner, 1961 - Eleusis) where the subject has a plate of grey tiles in front of him, and has to guess the picture after turning over the fewest tiles to reveal a black or white face, is a slow process, and encourages the subject to think rationally about the task.

This technique has been used to assess the amount of information in a picture by Attneave (1954), who asked his subjects to predict each tile before they turned it. Here a hypothesis testing mode was implied by the situation, and not invalidated by the results. The difficulty with the technique for our purpose is that to differentiate a rule-governed scan from a hypothesis testing scan, the rule governing the scan must be perceptible to the experimenter. A tile turning strategy of 'Turn any randomly selected tile' is rule-governed, but its results may not differ perceptibly from those of a hypothesis testing operation.

Most human pattern recognition is sub-conscious, requiring no active thought. In order to investigate this it would be well to have a process which allows subjects to act in a flow, rather than in discrete and spoken chunks. Moving pens admirably fulfil this.

In the moving pen technique of scanning research, the subject moves a pen which writes where the target picture is black, and either does not write or writes some different colour where the target picture is white. An example of this is "Magic Picture" childrens books, where some parts of a page are waxed so that a pencil does not write there. However one scribbles across the page, the pen only writes in the intended places, and the picture becomes clear.

The only experiment which has been done using this sort of technique is that of Podd'yakov (1970). He used treated paper and a chemical indicator, which left a trace on blank paper as well as a different trace on the target, thus entailing the difficulties of treated paper methods. His targets were outline drawings, and the subject's task was to trace along the lines. His report of the behaviour of subjects is that they scan at random to find some part of the figure, and change strategy on meeting a part. The two strategies used are to scan outwards in a fan-shaped path from the line segment already exposed or to scan at right angles to the apparent direction of the segment. He found that fan-type movements were much slower in continuing the line, although they were employed by 85% of his subjects.

He relates this behaviour to the micromovements of the eye when scanning. This analogy seems strained considering the vastly different amounts of information in the two cases. Podd'yakov alternatively speculates that this behaviour is that which would be expected of a predictive system working with cognitive concepts, as opposed to a simple edge following device at an automatic level.

In the terms of the present argument, the fan type movements are what would be expected of a hypothesis testing line follower, and only of that, 'guess the direction, then test it by moving in that direction'. The transverse scans might be feature analytic, and they are efficient. That fan type movements are prevalent, though slower, is an indication of the universality of the hypothesis testing mode.

This chapter has argued that pattern recognition is a major unsolved problem of perception, and that it is potentially soluble. Two models of pattern recognition have been advanced - hypothesis testing and feature analysis. These can be discriminated by observing the behaviour of the subject before recognition, conveniently by observing his scanning. A peripherally driven scan is indiscriminable from a hypothesis-testing scan, so peripheral information must be suppressed. Moving pens are the most promising way of doing this.

Chapter 2

CONSIDERATIONS FOR THIS STUDY

The previous chapter argued that it is only possible to discover whether a subject is using feature analysis or hypothesis testing mechanisms of recognition by investigating his behaviour before recognition. This requires the observation of scanning behaviour. Eye movements, the natural form of scanning, are difficult to interpret, as the previous chapter showed. Externally determined scan systems cannot rely on the subject to remember the image as he does for eye movements, but must store the result of the scanning and present all of it to the subject. This can conveniently be done for a single-celled eye scanning a black and white picture by having a pen which writes only where the hidden image is black.

Having set the study in context, this chapter will consider in more detail the essential concepts involved.

MODELS OF PATTERN RECOGNITION

Hypothesis Testing

The hypothesis testing model of pattern recognition asserts that the system makes an original hypothesis about the object and then checks whether the image is in agreement with the hypothesis.

The hypothesis is the selection of a pattern which is not only used to be checked against reality, but is also the result of recognition; the percept. Hypotheses that are grossly wrong will be rejected quickly and the hypothesis pattern and percept will change. Those that are correct will stand being checked until the world or the focus of attention changes.

A problem of the model is that the source of the original hypothesis is not specified. However, in a well-known world where the present circumstances are known, the original guess based on experience is likely to be accurate, eg, the butcher is easily recognised in his shop. In contrast, things or people in unusual environments are more difficult to recognise because the context leads to a wrong initial guess and several further inaccurate hypotheses will probably be made before recognition, eg, the butcher on holiday in Italy is less easily recognised. Thus a potential criticism of the model, the origin of the hypothesis, becomes a strength of the argument in explaining why recognition of the unexpected is more difficult.

The application of the model to some of the problems of perception is discussed below.

Recognition of most things visible is a necessary condition of any recognition model, and therefore does not provide a test of which model best fits reality. However, theories do differ in how adequate an explanation they provide of wrong or partial recognition.

One of the cases of false perception most difficult to explain by other theories of pattern recognition is the concave mask of a face, which is seen as a normal convex shape until the viewer is quite close to it (say 3 ft). The information for reversed depth from movement parallax is interpreted as movement of the mask. It appears to turn to face the viewer. The normal hypothesis, that it is a face which is convex, and that it can move to face the observer is strong enough to allow this illusion.

Hypothesis testing allows the non-visual evidence for the hypothesis, that is the experience of the faces being convex, to outweigh the visual evidence for concavity.

A further prediction of this model is that details of the object will become clearer as soon as one recognises the object, ie, brings the right hypothesis to mind. This is certainly common experience.

The theory gives a coherent account of alternating perceptions, such as those of the Necker Cube. Having one hypothesis "Cube this way out", the system keeps looking for corroboration of this in the image. It finds perspective cues which do not wholly agree - tries an alternate hypothesis "Cube that way out", and repeats, because the perspective cues never wholly fit either hypothesis.

Note that this last example has been of a 2-D drawing. The hypothesis testing model of perception is much more relevant here, where the information available from the image is less, and thus the amount of interpretation has to be greater, than in the 3-D case.

Feature Analysis

The alternative feature analysis model of pattern recognition asserts that detectors detect the features of the image, and then combinations of features and positions imply a certain pattern. A feature is something that can be detected by a simple detector, eg, the line detectors found in cats and monkeys. Other examples are bright points on the picture and their combinations, the curvature of black-white boundaries, loops and intersections, combinations of spatial frequencies, or, for letters, a template of each letter.

The term feature analysis has also been used in the specialised sense of a pattern recognition system where the features are detected and combined to give the result, but no information about their position is used. Gibson (1969) has postulated a model of this form for letter recognition. However, this is a restricted sense, and the words feature analysis are used in this study of models where feature detectors report the presence or absence of a feature at a position.

As for hypothesis testing, the feature analysis model must explain cases of wrong or partial recognition.

In real circumstances, detectors cannot report features as definitely present or absent, but only probably so. The pattern recognised is that with the highest combined probability even if its constituent features are ambiguous. The result will elicit the details from the stored representation that it calls up, and thus the perception of an indistinct object will be clear when it has been recognised.

The highest combined probability gives the result of recognition, but below some minimum probability the result is taken to be indistinct. This minimum combined probability would be greater for an unlikely result than for a likely one, so that things in unexpected places take more information to be recognised.

If one considers the Necker Cube, it provides two sets of features, both true with high probability. The observer then sees one or the other, rather than an amalgam of both. The model needs other additions to explain the alternation.

Feature analysis and hypothesis testing agree in predicting that most things will be correctly recognised. Hypothesis testing appears to provide a better explanation of many illusions, but feature analysis can be extended to explain many of them, so that it is difficult to differentiate the theories on the grounds of wrong recognition. Only hypothesis testing provides an explanation of the concave face illusion, but this single case cannot be taken as evidence for all recognition.

However, the two theories do predict different behaviour to gather information before recognition. The feature analytic model, having no cognisance of what is out there, must rely on some external search operator. The hypothesis testing model will search in a way that tests the current hypothesis.

The output from a feature analytic system viewing a picture which is not adequate for recognition is that something is there, but not recognisable. The hypothesis testing system reports that it is a widget, but not properly checked. Thus further scanning can be directed by the hypothesis testing system to parts of the image which carry most information about widgets, but the feature analysing system must rely on some information other than the result of recognition to guide the scan. This can either be peripherally driven or a regular scan, as was argued for eye movements. The single-celled eye allows no peripheral driving, and so only a regular scan can result from a feature analysis system.

The hypothesis testing process is thus essentially cyclic, "guess then check, then change, guess and recheck". It allows the guess to be combined from simple concepts, and be modified, to build up a more complex percept, such as "A blue Mini carrying a brass bedstead". It embodies continuous refinement of the guess, and concentrates this refinement on the areas where the guess diverges from the observed reality.

The feature analysis process is essentially parallel, with all feature detectors and combination detectors working continuously and simultaneously.

Internal Representation and Hypothesis Testing

The concept of hypothesis testing is independent of the internal representation of the object, ie, the pictorial representation within the pattern - for Fido, the representation of his appearance. The basic notion of hypothesis testing is that the subject acts to acquire information, and changes his acts on the basis of the information he has. This is to be contrasted with feature analysis, where the subject acquires enough information to decide, before changing his acts to take account of what he knows.

Consider three possible representations of letters; feature sets, relational structures and fourier transformed feature lists; and what they imply for the behaviour of the subject in this situation.

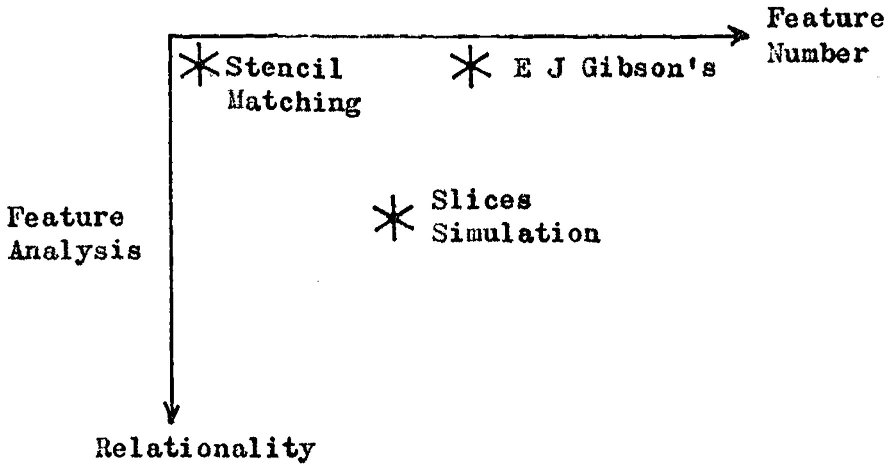
Feature sets in the non-relational form, assert, for example, that an "E" has 3 horizontal bars, a vertical bar, and no loops. There must be at least enough relation information to assert that these are connected; non-connected bits are part of another letter. Hypothesis testing demands that after detectors have found a number of features which are true of the present image, a matching of sets will reveal those letters that are compatible with the information gained so far, and also the information needed to separate the possibilities. Thus the existence of 3 horizontal bars and one vertical is compatible only with "E" or "B". The information needed to discriminate is the existence of a loop or line ends - a loop implies a "B", a line end implies an "E". If the matching of sets is arranged to report only successful recognition or failure, then it can only support feature analysis.

Relational descriptions of letters hold letters as collections of facts like '"O" is a loop with no sharp corners on the left'; '"Q" is like an "O" but has a bar through the bottom right'. This reveals one obvious way in which inadequate information can help guide the search for fuller information: having triggered a "same" part of the description, (the loop of an "O"), look where the "different" ought to be, (the bottom right for the tail).

Once again, variants can be conceived which do not produce any useful guidance when fed with inadequate information. The description of relational descriptions given here is simplistic; letters do not seem complex enough to warrant such a powerful method, whose special merit is that new information is assimilated easily, and that complex bodies of information do not occupy huge amounts of storage.

A diagram of the relation of internal representations and recognition models is shown in Figure 2.1.

Fourier transformed representations of letters are discussed here as representatives of whole picture transformations. In a Fourier transform of a picture, the intensity of one component, the product of an orthogonal pair of spatial frequencies, is represented as one point in the Fourier plane. Thus a property of the whole picture is mapped to a point, and a property at a point is mapped to the whole plane. Imperfect knowledge of the image converts to an imperfect Fourier transform. Letters can be stored as representations of Fourier transforms of their shapes.



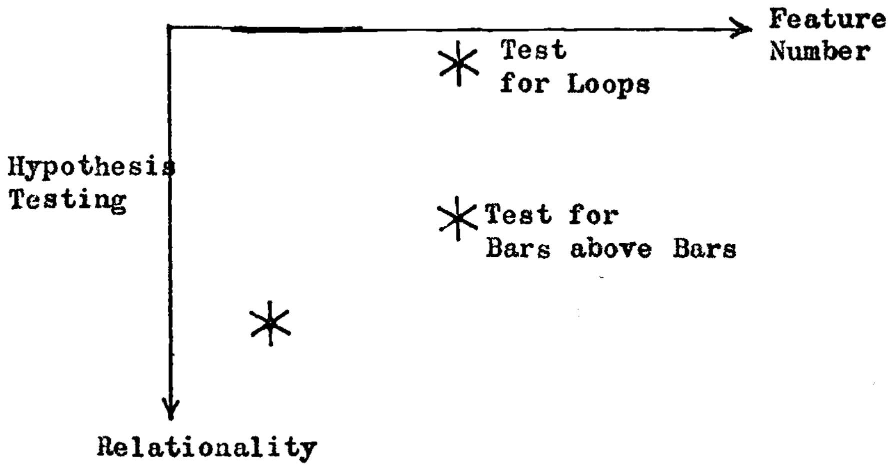


Figure 2.1

A Pattern Recognition model can be considered to occupy some point in this space, with dimensions;-

1) Number of features noted in each exemplar

2) Relationality - how much relative positions matter

These two dimensions depend on the internal representation.

3) Feature analysis - hypothesis testing. This is a dichotomy rather than a continuum, and is independant of the internal representation.

The problem of recognition remains, because a Fourier transform is still spatially organised. Either pattern recognition model can be applied to it. If this recognition process is hypothesis testing, then the guidance it gives for further acquisition of information will be in terms of spatial frequencies, not positions. The meaning of this for normal search behaviour is unclear. It seems likely that the search behaviour would be an area-covering scan, and scanning would not differentiate the two models. It seems probable, however, that the spatial frequency analysers found in people (Davidson, 1968) are used for surface texture perception, not for shape analysis.

Experimental Considerations

An activity has a cost, which may be in money, as in some card guessing games, or in effort. The concept implies that people will attempt to attain their goals with minimum expenditure, by economising on costly activities. Applying this concept to recognition, two costs may be distinguished - the cost of acquiring the information, which may be taken as the time and effort needed to search for it, and the cost of thought, that is, the time and effort involved in processing the information.

In a recognition task which can be done by either hypothesis testing or feature analysis, hypothesis testing will use the least information, because it can use all the information available to it to choose the most effective place to seek more information. If the cost of information is greater than the cost of thought, then hypothesis testing should be preferred.

Correspondingly feature analysis will be used where information is adequate, revealed at the same time and thus cheap relative to processing, because feature analysis uses more information and less processing than hypothesis testing.

Generating a new pattern involves a capital cost - low for hypothesis testing, only another store entry, and high for feature analysis, another set of detectors. Feature analysis is therefore best adapted to recognising common objects, since it involves a high cost to set up a pattern, but a low cost to continue to use it. Hypothesis testing is better adapted to rare situations, since it involves a lower cost to set up a pattern, but a higher cost to use it.

A recognition system using both models of recognition could have the best of both. Hypothesis testing is probably better adapted to recognising the novel or unusual, feature analysis to recognising easily. If one novel input comes to be needed to be recognised often, as with letters in learning to read, then a feature analysis machine for it can usefully be built. Feature analysis can be conceived as applying the decisions of hypothesis testing in parallel - that feature analysis embodies in hardware the decisions hypothesis testing takes with software. Making the distinction between the two models in this form emphasises the futility of trying to decide between them after recognition or mis-recognition. The discussion of relative costs of information and processing suggests that an experiment, such as this one, intended to reveal hypothesis testing if it is available, should use expensive information.

Hypothesis testing would also be preferred if the test was to match items, not to identify them - that is to say that they are the same or different from a known exemplar, rather than to say what they are. A matching task means that there is no problem of finding an initial hypothesis.

Thus a task where a subject scans, and thus acquires more information at a cost to himself, and one where he matches a target, is likely to be solved using a hypothesis testing mode of pattern recognition, if it is available.

A task which embodies this requirement is scanning with a detector that reports only black or white at one point at a time - a single-celled eye. It incorporates scanning, allowing decisions made before recognition to be observed, and makes the information from scanning expensive because it is slow and a nuisance to collect.

Equipment

The ideal embodiment of the task would be a pen which wrote only where the hidden image was black. This can be done, as in childrens "Magic Picture" books, by treating some parts of the paper with wax so that the pen does not write there or writes a different colour. This method allows natural movement of a simple pen, but leaves a trace on the paper which may be visible to the subject. It also gives information about the slope at which a scan line crosses a picture edge.

The embodiment chosen for this task had a pen moved around the paper by an external joystick, with its writing controlled by a flying spot scanner.

The experiment is to be set up to show whether subjects can use a hypothesis testing method of recognition, under the most favourable conditions for it. If this experiment shows that hypothesis testing can be used, then studies should be undertaken to show when it is used: these are pointless if it is not used.

Targets for recognition

The choice of targets for recognition is between having unknown patterns, such as the Edmondson shapes (Figure 4.2), or well-known targets such as letters. The unknown targets can be designed to various forms of scan, but have the disadvantage that to be matched they have to be shown to the subject, probably on a sheet. Thus the subject has a task of searching a list of specimens as well as that of producing an adequate delineation with minimal scanning: this biases him towards scanning more and thinking less, because the cost of recognition has been raised by the effort of searching a list.

Known targets, such as letters, can be compared with an internal representation: this, by reducing the cost of recognition, biases the subject to thinking more and scanning less. One counterindication to the use of letters is that the internal representation of letters is presumably well adapted to recognising the normal simultaneous presentation of a letter, and may not be easily adaptable to controlling a sequential input, as in the present experiment. A new pattern may avoid this by being internally represented in a less specialised way, and thus elicit hypothesis testing more easily.

One difficulty with the task is that the lines on the paper are not the lines of the letter, Figure 2.2. The lines on the paper-scan lines - begin and end on the edges of the lines of the letter, perhaps better called letter bars. If hypothesis testing

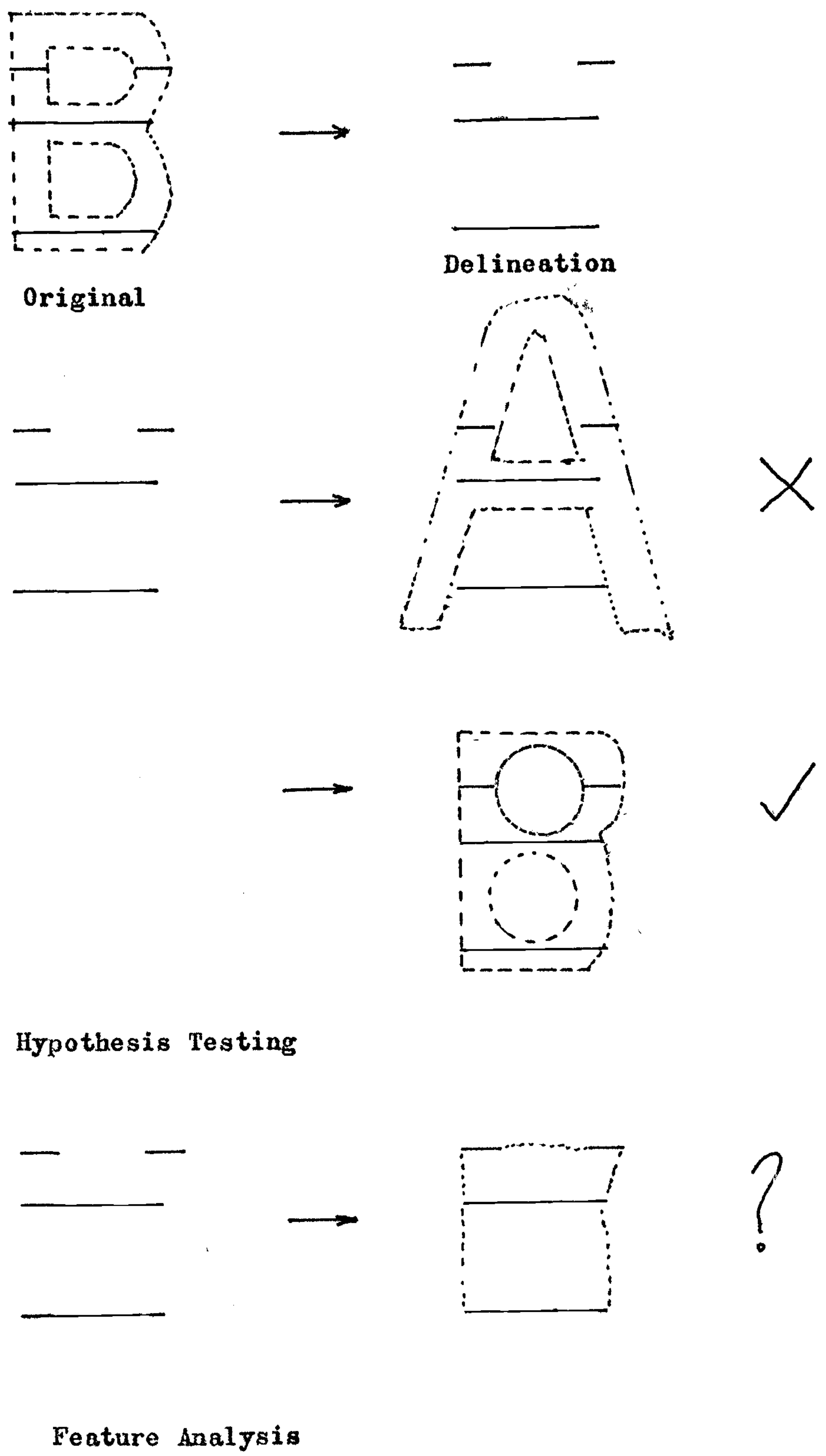


Figure 2.2
Shape completion

is used, there is not much problem; the question of whether the scan lines are compatible with the hypothesised letter is not different in principle, nor more difficult than deciding shapes from fractions of themselves.

If feature analysis is used, the problem is more complex. There is no apparent algorithm which will outline the shape of the letter, given the scan lines, unless the scan lines are closer than the width of a letter bar. In this case joining nearest ends of scan lines will outline the letter. Thus it might be expected, on these grounds, that a feature analysis system will be most efficiently informed by a uniform line scan of this spacing. Corners in the scan lines will cause some degradation, as leaving ambiguous the difference between a corner and an end, and also introducing a confusion in their own right.

Summary

This experiment is designed to differentiate between two forms of pattern recognition, designated feature analysis and hypothesis testing.

A subject using feature analysis needs an image to recognise. If the image is not recognised, then nothing is known about it. In particular, if the image is not recognised he cannot work out what knowledge would help him to recognise it. This is true of, eg, a straight-through Perceptron type recogniser, which produces only the information that nothing is recognised. The critical part of this system is that the subject cannot control his intake of information on the basis of what he already knows, when this is inadequate for full recognition.

A subject using hypothesis testing has an idea of what is being recognised, and acts to prove or disprove this. Thus he is able to use this internal hypothesis to guide his information uptake, and will change his hypothesis when incoming information proves it invalid.

The use of feature analysis or hypothesis testing will be affected by the relative costs of information and processing, and by the rarity and probability of the object. It will not be affected by the form of the internal representation of the object.

The use of hypothesis testing can best be investigated by having the subject scan a letter with a single point, a situation favourable for hypothesis testing, and where its effects are detectable.

Chapter 3

EQUIPMENT

The equipment needed for this study was a moving pen which wrote only where the hidden image was black. This was done with an X-Y plotter, whose position was controlled by a joystick or a scan generator, and whose writing was controlled by a flying spot scanner inspecting a transparency. The layout is shown in Figures 3.1 and 3.2, and a typical output is shown in Figure 3.3.

Flying Spot Scanner

The flying spot scanner shown in Figure 3.4 was based on a Telequipment Type S51T single beam oscilloscope. The timebase was switched out; the X-component of the scan was applied to the X-deflection amplifier and the Y-component drove the Y-amplifier. The field size was defined by standard positions of the oscilloscope gain controls, and by clamped external potentiometers, to be a square. Its position was centred on the screen using the oscilloscope position controls.

Transparencies were placed directly on the tube face. A photocell was supported 25 cms from the tube face, together with its first stage of amplification. This distance was the shortest at which a black object in the centre of the screen could be discriminated from a clear object at the edge of the screen.

Early trials used a simple DC coupled photocell to detect the oscilloscope spot. This necessitated a light-proof box round the scanner and a very bright spot. The resulting spot width reduced the resolution of the scanner and damaged the tube phosphor when it was left stationary; the box was inconvenient to operate, and on occasion concealed mishaps such as the target

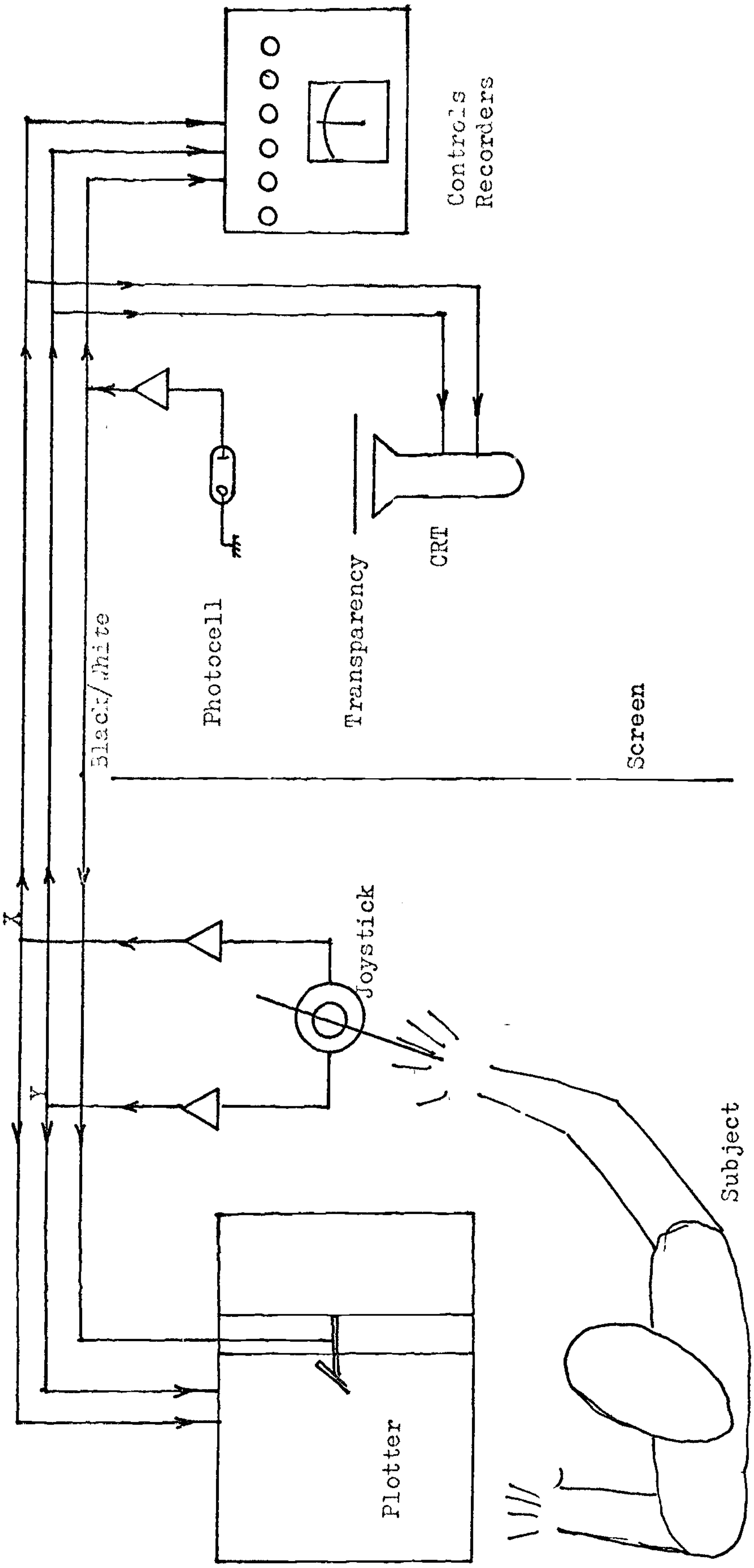


Figure 3.1

Schematic layout of equipment

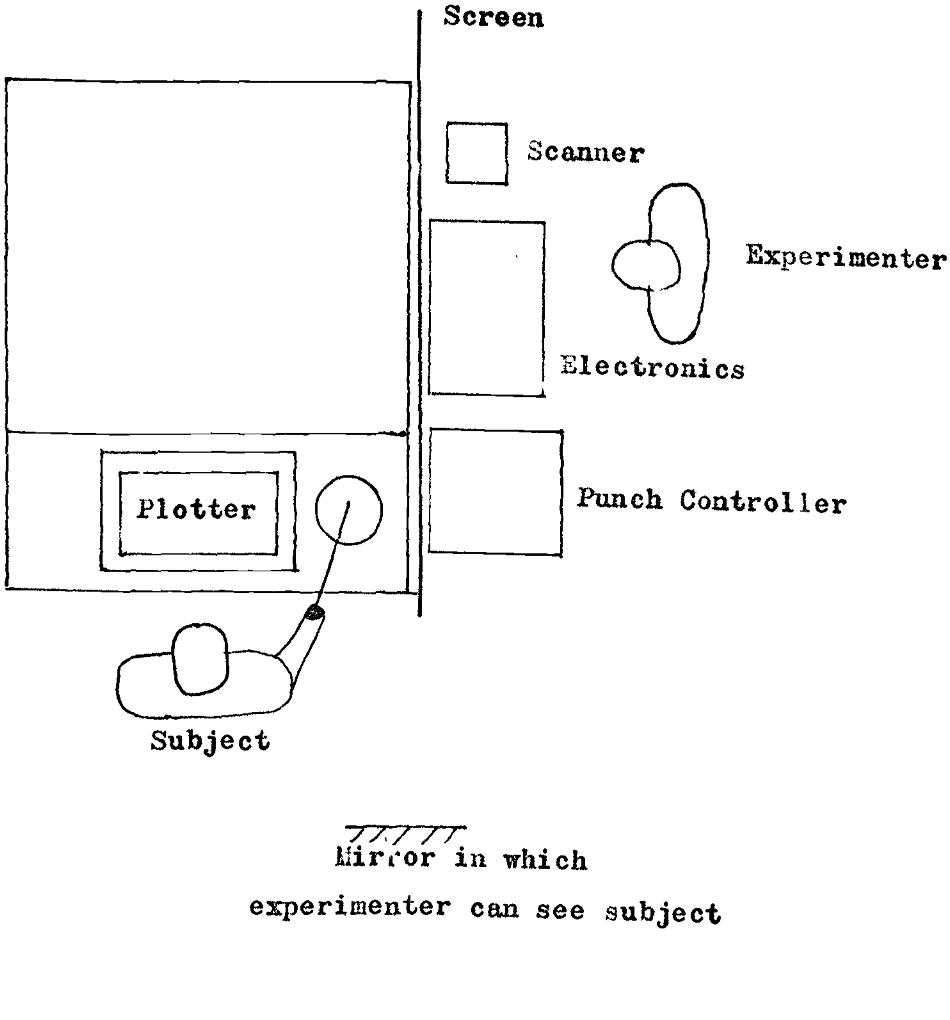


Figure 3.2
Layout of Equipment

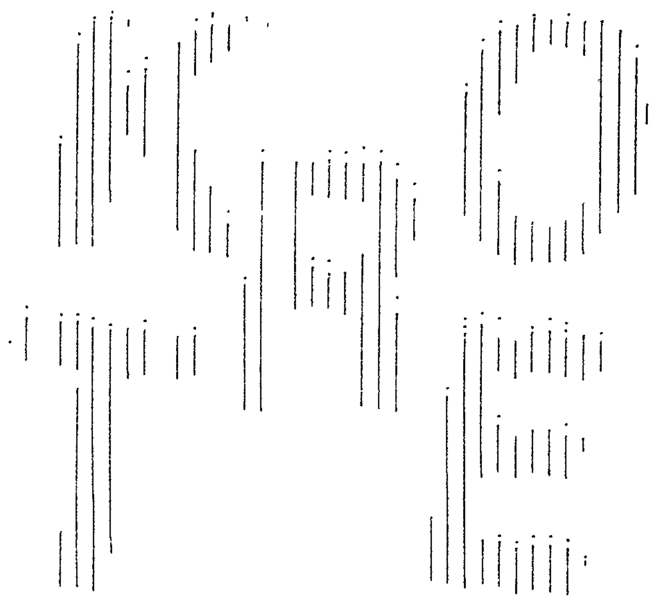


Figure 3.3
Typical Output

shifting or curling in the heat.

In order to operate the scanner in the open with a small spot, a phase sensitive detector was built. The electronic circuit diagram is shown in Figure 3.5. The spot brightness was modulated at 50KHz nominal, 43.4KHz measured, and the output from the photocell was synchronously rectified, thus preserving only the component at that frequency. The frequency was chosen as the highest frequency at which the oscilloscope, photocell and amplifier produced a useful response.

The system had a response delay of 0.7 milliseconds; it was unaffected by normal room lighting, but had to be shielded from strong direct sunlight. It detected lines narrower than 1mm over an area 11 x 8 cms. The overall performance is perhaps best demonstrated by Figure 3.6, which is the output on the plotter for a target of 1mm lines on a 10mm rectangular grid. The obvious rhombus distortion is due to interaction between the oscilloscope amplifiers. It has been ignored thoughout the experiments since its effect on letters is only to change them from an upright to an italic typeface, for example

E → *E*

Paper Tape Punch and Analog to Digital Converter

The 8-hole paper tape punch, Teletype Corporation type BRPE, and its drive electronics were available from a previous experimental device. The electronics were modified to accept

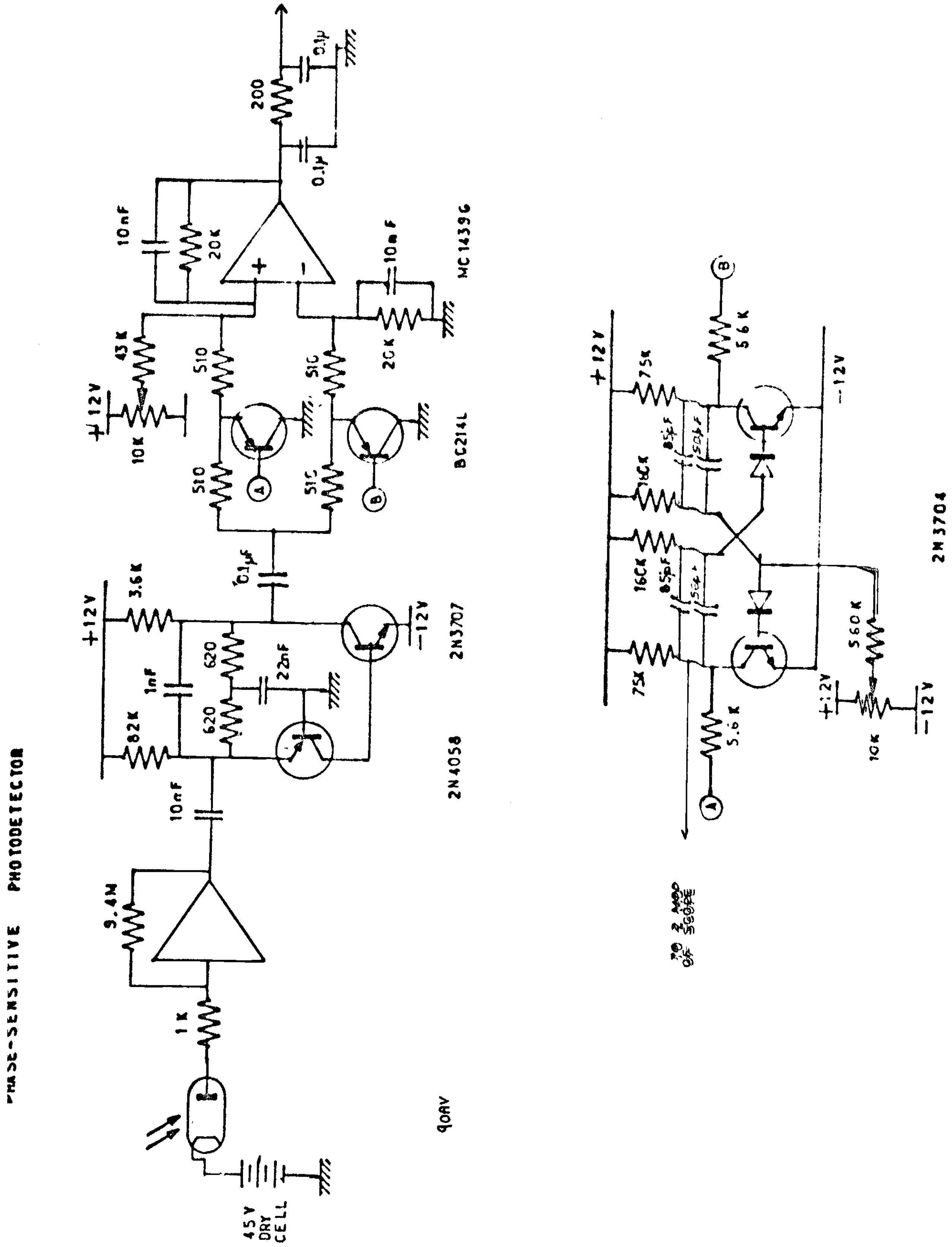


Figure 3.5
Flying Spot Scanner

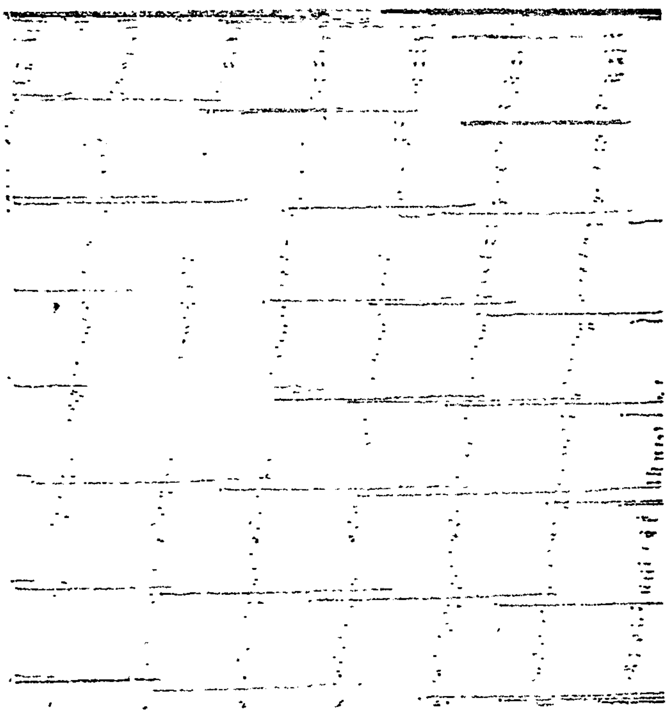


Figure 3.6

1mm lines on a 10mm grid

TTL logic inputs. The punch, being very noisy, was mounted on a vibration damper outside the experimental room.

An A-D converter, whose electronic circuit diagram is shown in Figure 3.7 was built using TTL logic. A sampling cycle is initiated every 0.1 seconds nominal (0.0997 seconds measured) by an astable multivibrator. Thereafter, within the cycle, timing is dependent on the punch, which produces a signal when it is ready to accept a new input. On successive punch cycles the X and Y levels are compared with a staircase waveform, whose generating counter is stopped at equality. The number in this counter is taken to be the digitisation of the level. 8 bits resolution was used (1 part in 256). The brightness was punched at the third cycle on track 5, leaving the other tracks blank. This allowed the analysis programs to check that they were maintaining synchrony with the punch, since every third number, the one which represented brightness, could only be 0 or 16. Controls were arranged for the experimenter to stop the tape or to punch blank tape.

Joystick

A simple joystick was built using two standard plastic potentiometers. This device felt quite pleasant to use as the potentiometers were filled with viscous grease, and more sophistication was felt to be unnecessary. The joystick was always placed conveniently for the preferred hand of the subject. The potentiometers had to be replaced on occasion as they wore out, but otherwise gave no trouble. The electronics are straightforward: two operational amplifiers were used to convert the signal level to the standard for the apparatus. The centre

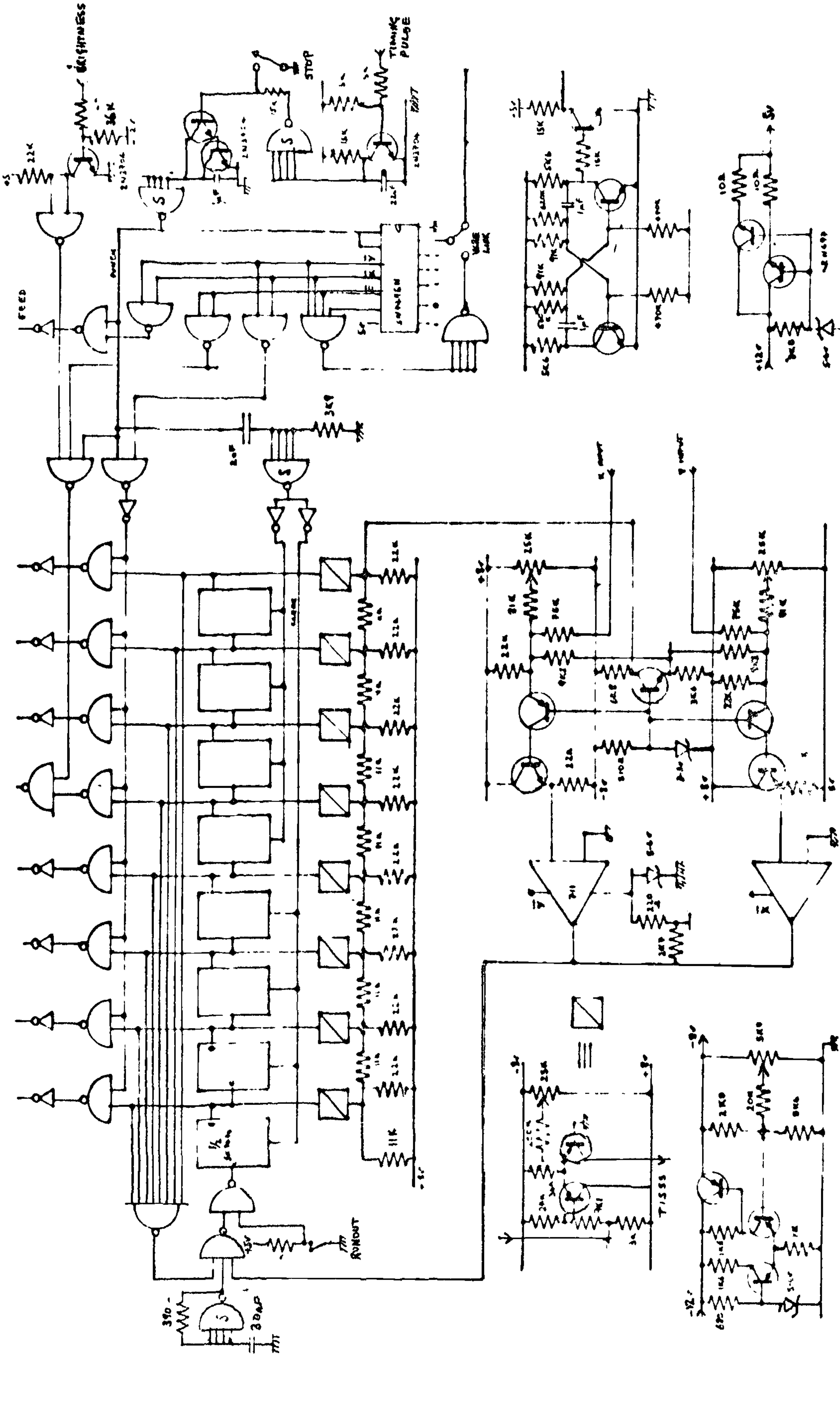


Figure 3.7

Analogue-Digital Converter

point of operation was adjusted by shifting the potentiometers on their mountings. The amplifier gains were arranged to give pen movements of approximately the same amplitude as movements of the joystick.

The circuit diagram of the electronics is shown in Figure 3.8.

Line Scan Generator

The circuit diagram of the electronics is shown in Figure 3.9 and the output in Figure 3.10.

This scan generator produced a slow version of the scan used in a television set, or in reading. One integrator-comparator loop, arranged to have different rates of rise and fall, was coupled to another integrator-comparator loop, so that the second was augmented during the flyback of the first. The flyback of the second loop started when an upper trigger point was reached, and interrupted that of the first loop. The rates were arranged to give a good writing speed on the X-Y plotter. During either flyback the pen was inhibited from writing.

The resulting signals were output as the TV(X) scan, and also inverted and interchanged to form the TV(Y) scan. This last was arranged to proceed contrary to the natural ordering so that the plotter bridge might not obscure the line it had just drawn.

The gain control was adjusted to give six lines per frame.

Square Spiral Scan Generator

The electronic circuit diagram is shown in Figure 3.11 and the output in Figure 3.12.

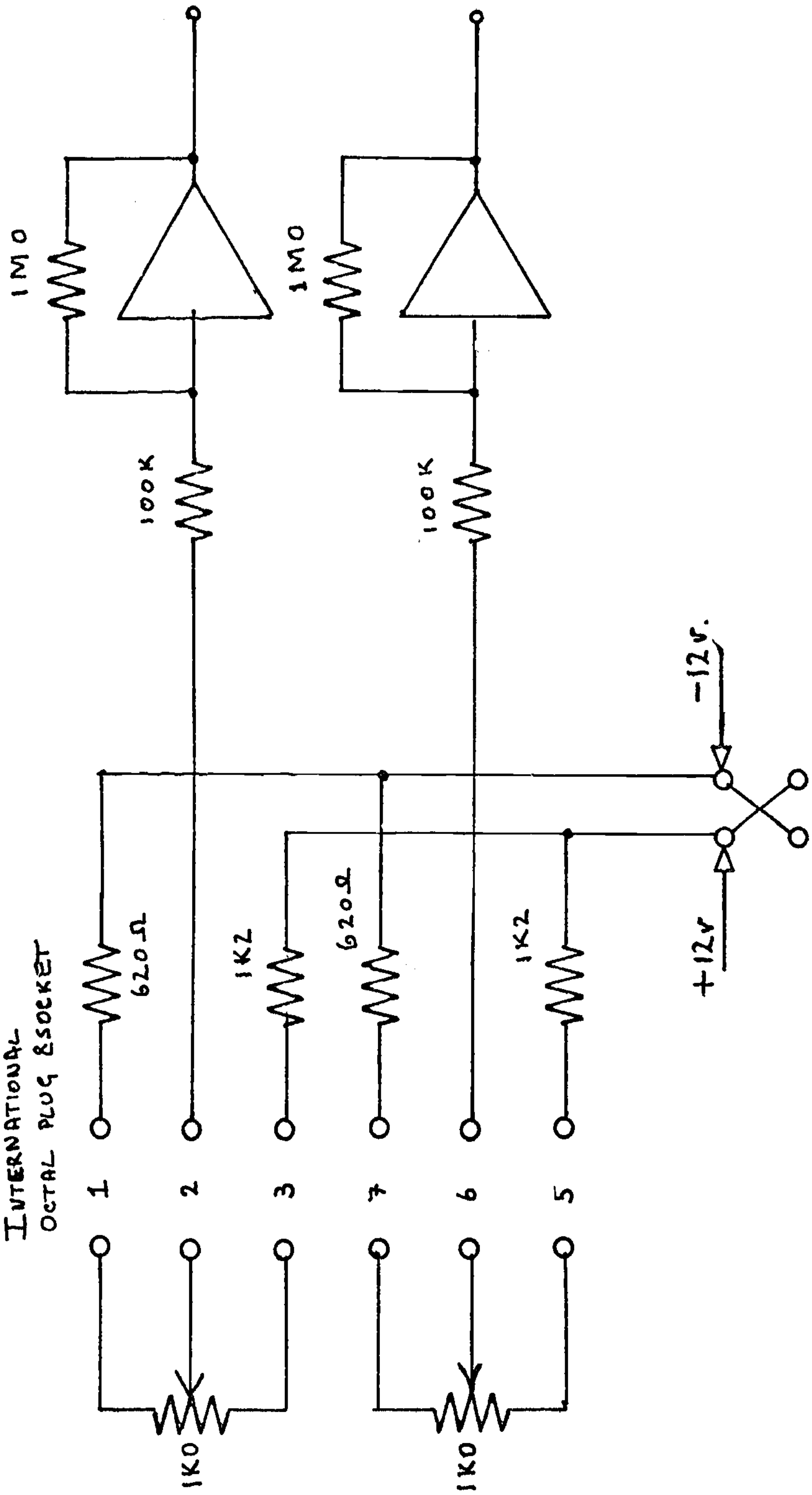


Figure 3.8

Joystick Amplifiers

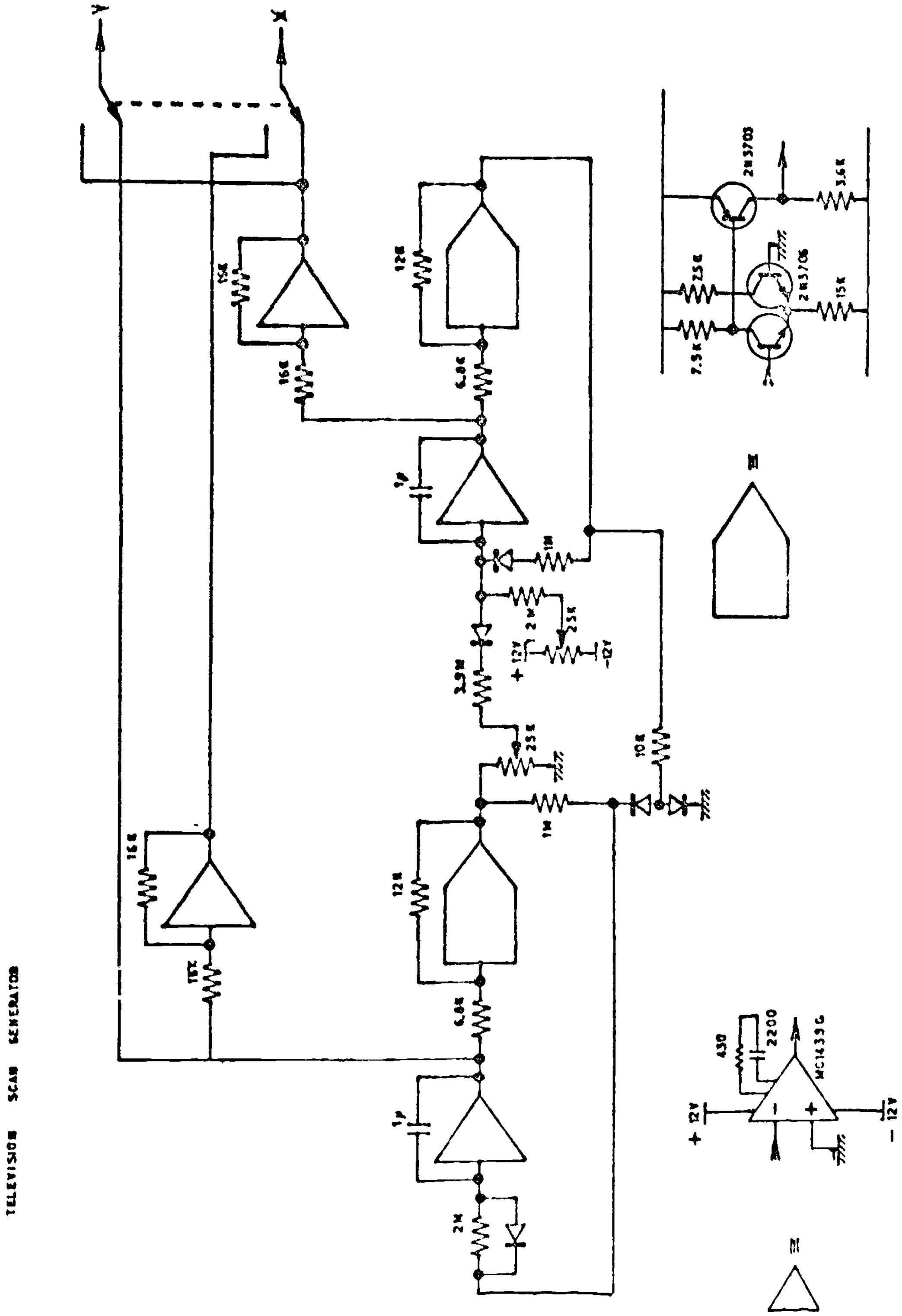
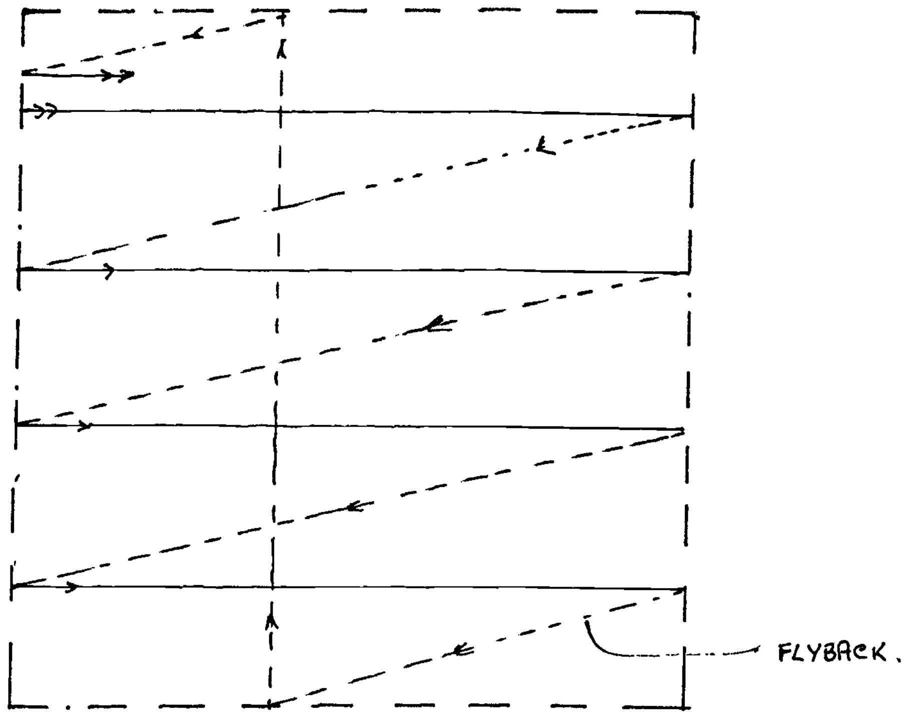
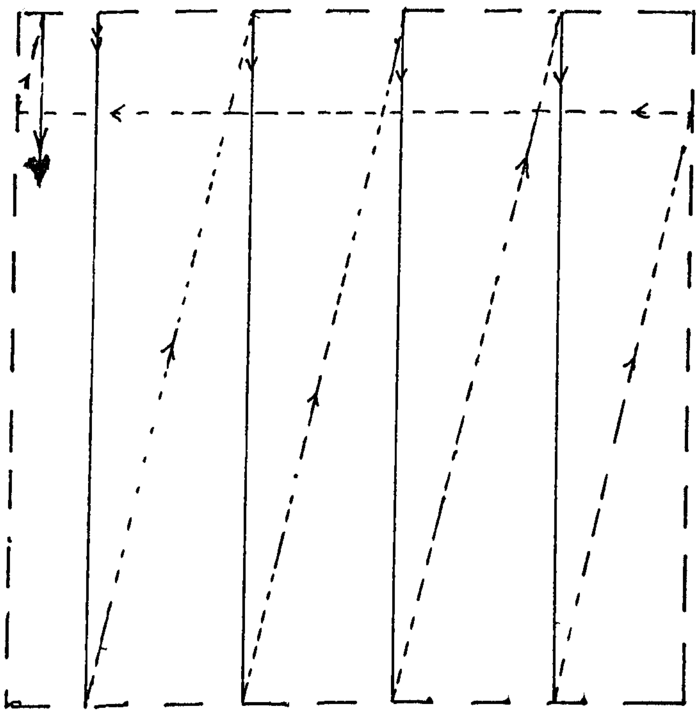


Figure 3.9

Line Scan Generators



TV(X)



TV(Y)

Figure 3.10

Television scan patterns

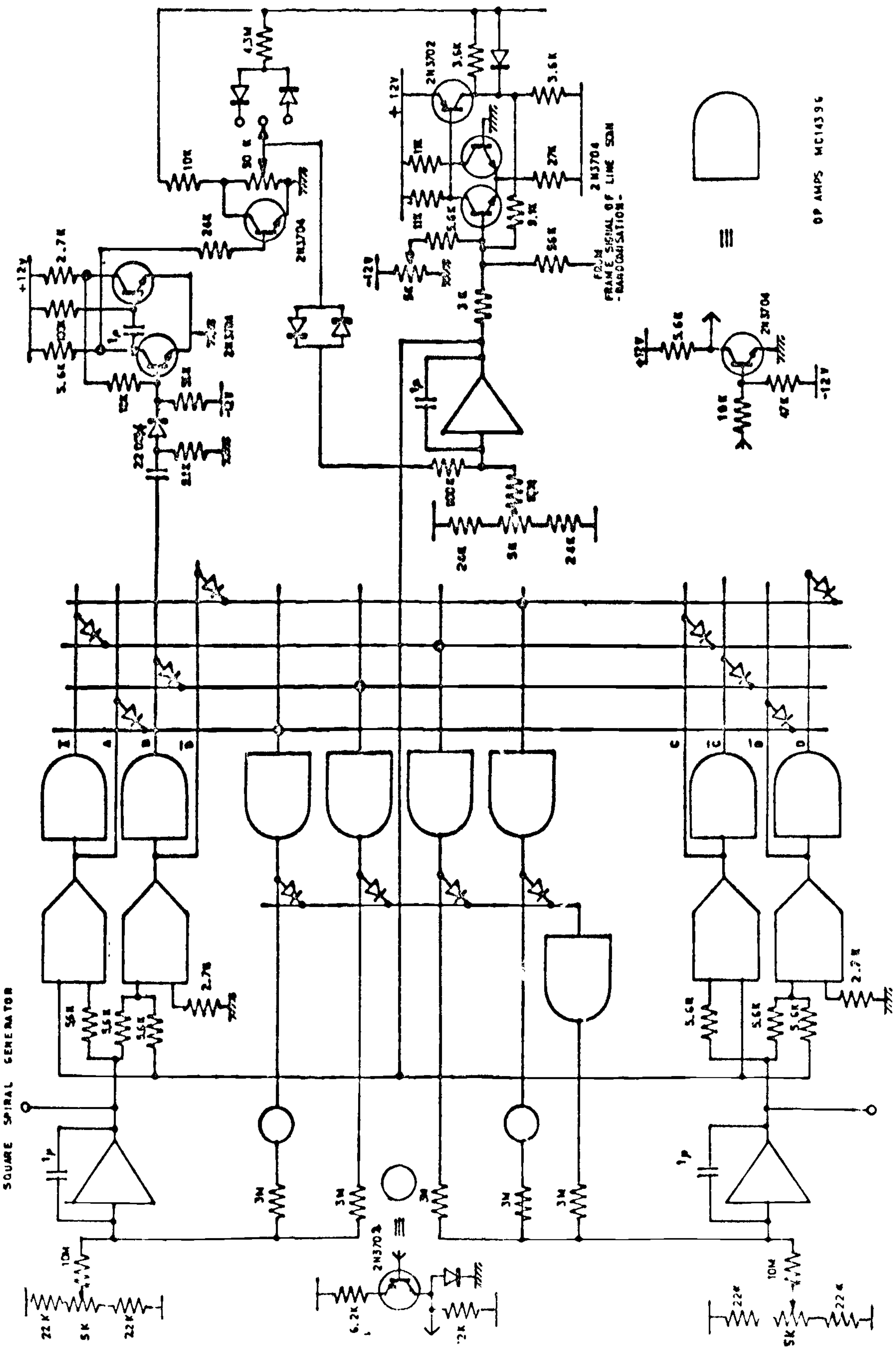


Figure 3.11

Square Spiral Scan Generator

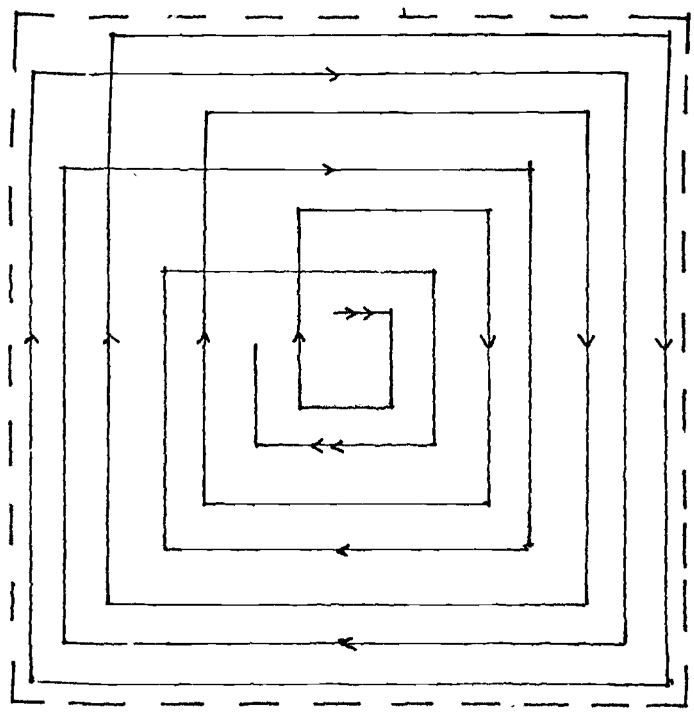


Figure 3.12
Square Spiral Scan Pattern

Two integrators, each with two comparators, and some logic circuits, were arranged so that first the Y integrator went down until it was equal to the comparison level, and then the X integrator went down until it too hit the lower limit. The Y integrator was then made to go up, followed by the X integrator. Thus the pen driven by the pair of integrators had been made to go in a square path. At the top right hand corner, where both X and Y were at the maximum, the trigger level, set by a master integrator-comparator loop, was incremented or decremented for a time set by a collector-coupled monostable. The master integrator-comparator loop, which set the trigger level for the rest, switched from expansion to contraction or conversely at a point varied slightly by an input from the line scan generator, so that the next cycle should not go through exactly the same points.

The gains were arranged to give about 1cm separation between adjacent lines.

Diagonal Generator

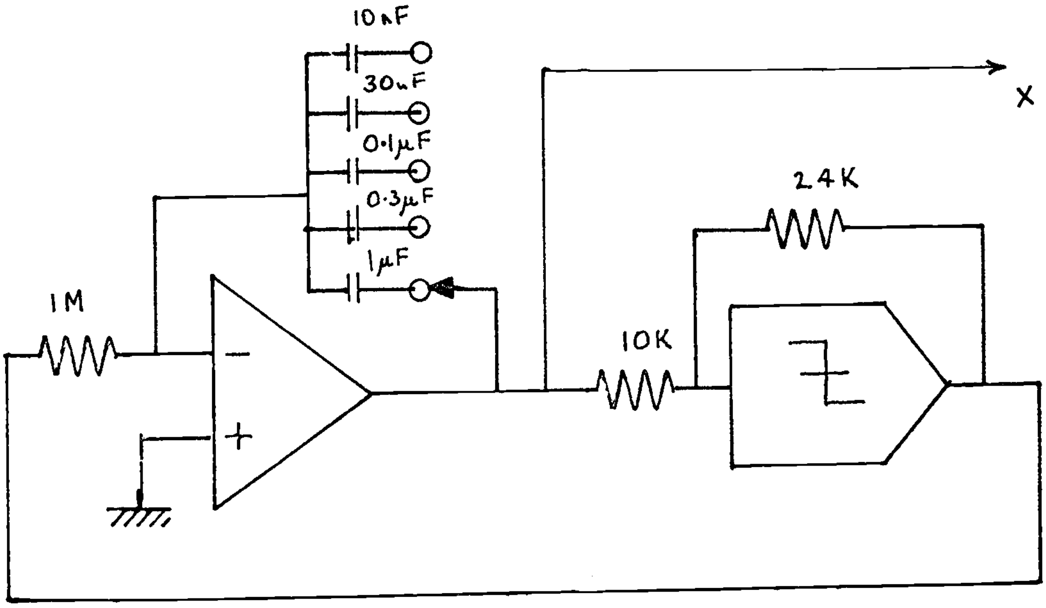
This was realised using two integrator-comparator loops working independently. The effect is that of a frictionless billiard ball launched at 45^{o} to the sides of a square, perfectly elastic table.

The electronic circuit is given in Figure 3.13 and the scan generator in Figure 3.14.

Round Spiral Scan Generator

The electronic circuit is given in Figure 3.15 and the output in Figure 3.16.

The circuit is based on a loop of two integrators and an



Repeated for the "Y" channel

Figure 3.13

Diagonal Scan Generator

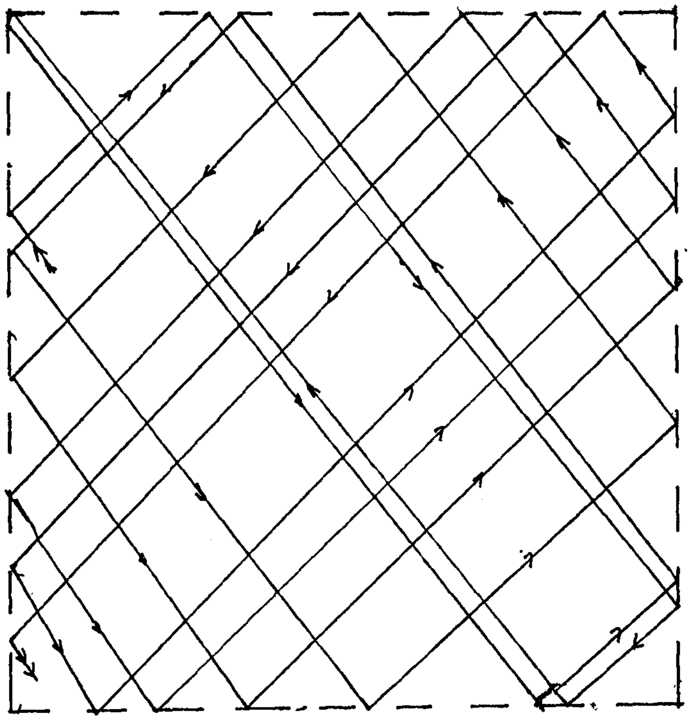


Figure 3.14
Diagonal Scan Pattern

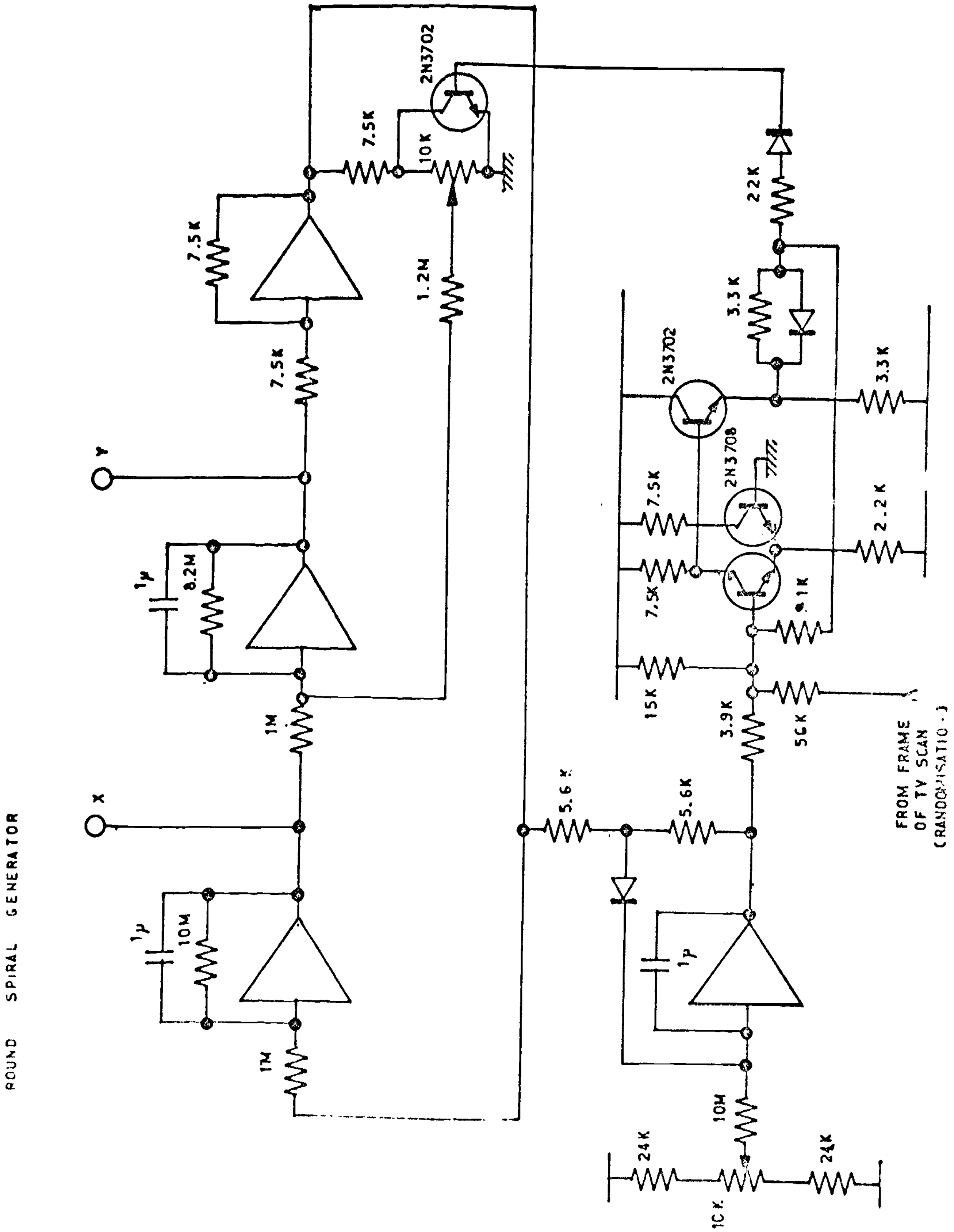


Figure 3.15

Round Spiral Scan Generator

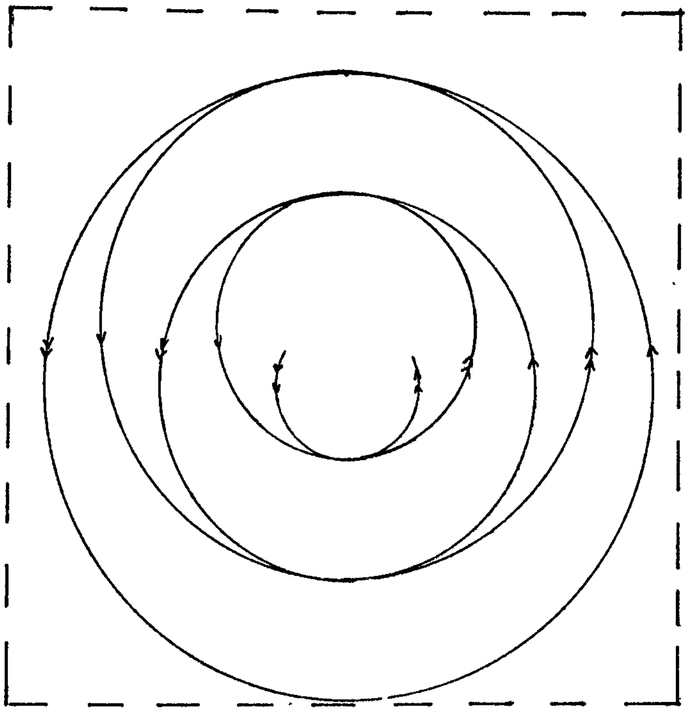


Figure **3.16**
Round Spiral Scan Pattern

inverter, forming a slow sinusoidal oscillator with two outputs in quadrature. The basis of the circuit is described by Korn and Korn (1964). The loop is arranged to have a constant gain in amplitude, and a loss of twice that value is controlled by a biassed comparator of high hysteresis working from the peak value of the oscillator.

This method of achieving this scan pattern has the disadvantage that the pen moves at constant angular velocity. Any speed setting thus gives a range of speeds from excessively slow at the centre to excessively fast at the periphery. The fast speed entails bad definition.

The rate of increase of radius was adjusted to give about 1cm separation between adjacent lines.

Switchboard

The outputs from the various scan generators and from the joystick were brought together at a switchboard, allowing the experimenter to select one of them for a trial. The selected output was passed through a limiting amplifier, constraining the pen to move within a square. Without this the subject tended to explore the mask of the oscilloscope screen, which operated the pen in the same way as a black target on a clear background.

The electronic circuit diagram is shown in Figure 3.17.

X-Y Plotter

The subject's view of the plotter is shown in Figure 3.18.

The X-Y plotter used was a Bryans 20170, equipped with a fibre tipped pen which gave more reliable writing, a slightly bolder line and needed less attention than the original capillary pen. Red, green, and blue pens were used indiscriminately.

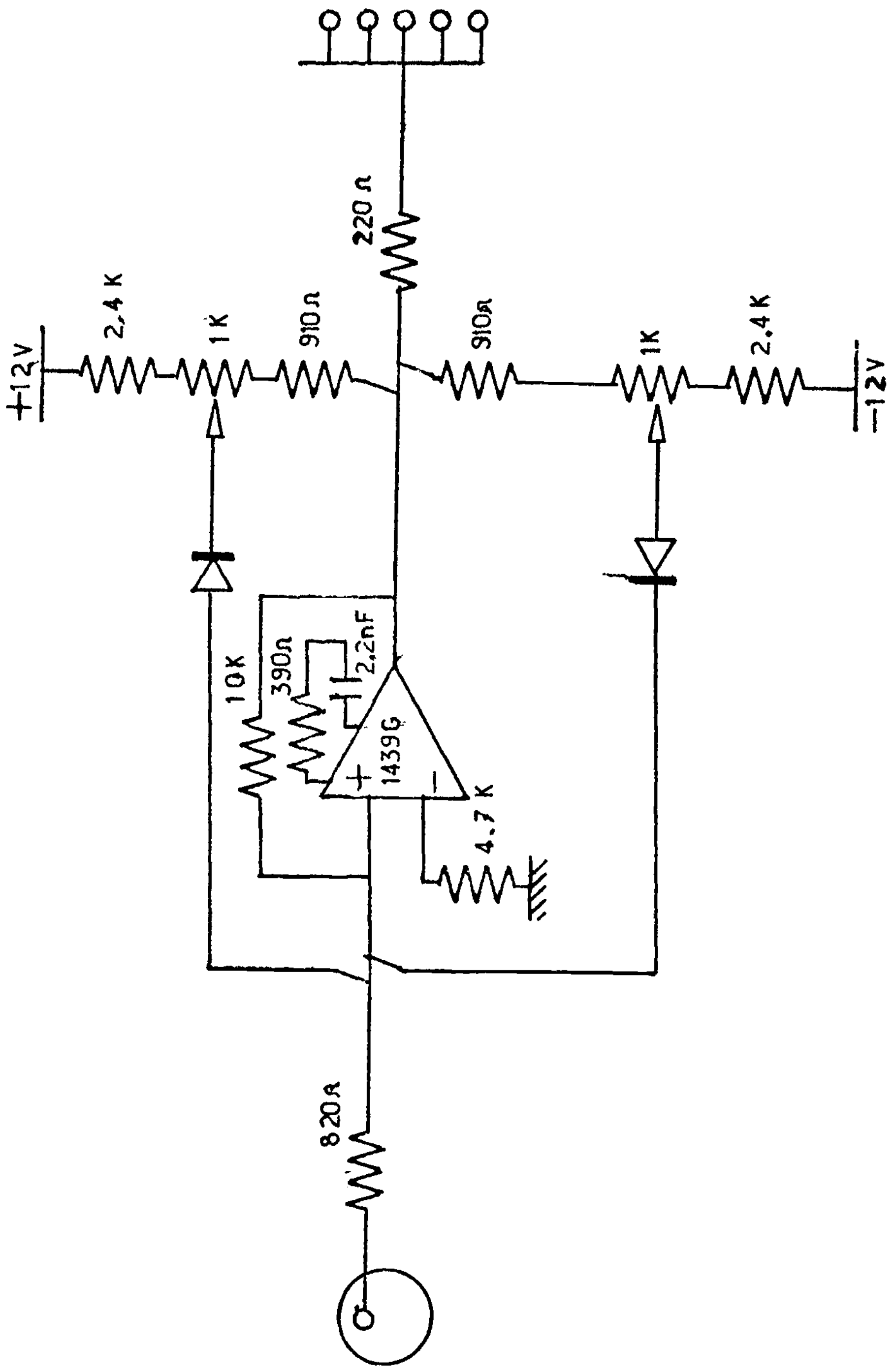


Figure 3.17

Scan Distribution Amplifier and Limiter

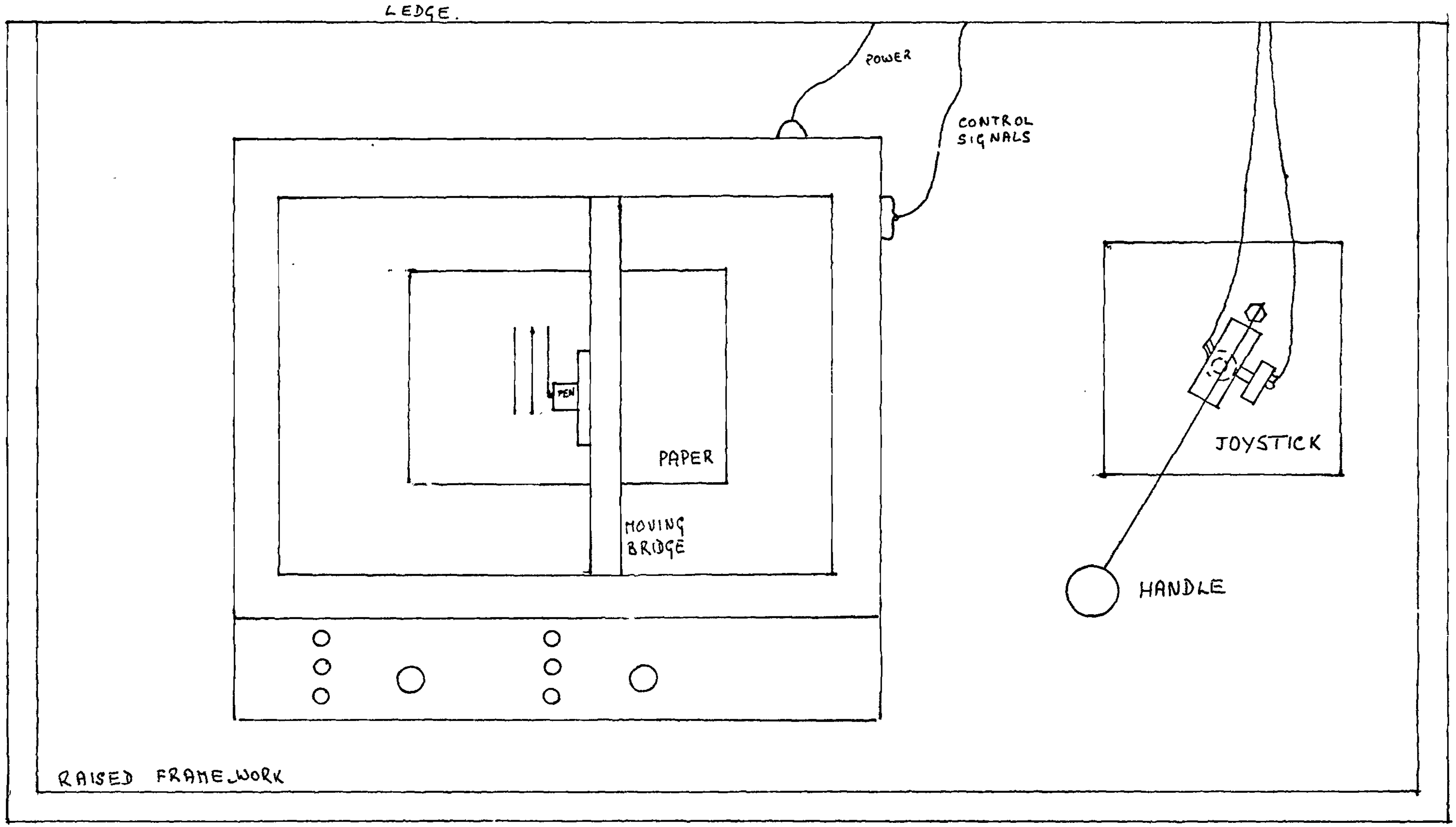


FIGURE 3.18

SUBJECT'S VIEW OF X-Y PLOTTER

The pen recorder amplifier sensitivities were pre-set to give a size of picture, 9cm square, shown in Figure 3.3. The unit, apart from routine cleaning, operated reliably and consistently.

A transistorised switch was arranged for the remote pen lift. A set of controls was arranged that turned on the plotter, started a timer, and started the paper-tape punch in one operation, (Figure 3.19). Another operation turned off the plotter, stopped the timer, and stopped the paper-tape punch after running out a foot of blank tape. This was worthwhile because these operations had to be done for each trial.

The Analysis System

The analysis of the tapes produced during the experiment was performed by a set of computer programs, written in POP-2 programming language (Burstall et al 1968). The analysis was produced in two runs. The first run converted the data on the tapes into a line-printer picture of each trial, and calculated some basic descriptive statistics for each trial. These were:

1. Distance moved. The sum of the distances between successive sample points. This was the total distance through which the pen moved during that trial, measured in arbitrary units defined by the A-D converter. The arbitrary units corresponded to 0.7 mm on the scope face and 0.9 mm on the plotter. The letters used in the paradigmatic experiment were 35 arbitrary units high. This was the quantity the subject was asked to minimise.
2. Writing distance. The sum of the distances between successive sample points if both were black. This was the distance over which the pen wrote during the trial,

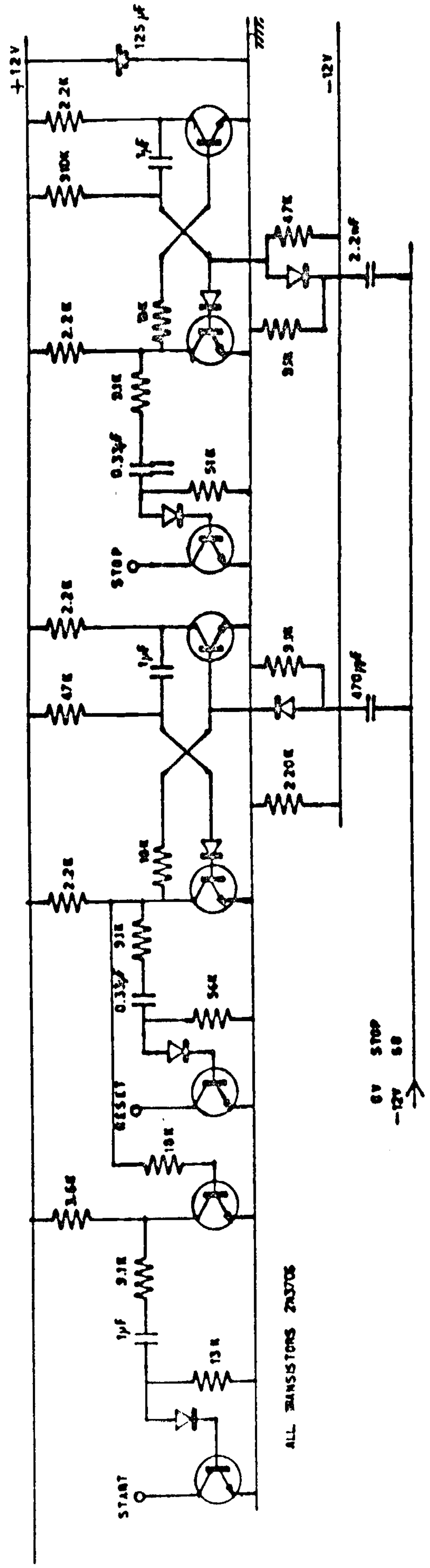


Figure 3.19

Controller

measured in the same arbitrary units as the distance moved.

3. Time. The number of sample points taken, which is the duration of the trial in units of 0.1 seconds.

4. Crossings. The number of times one sample point differed in brightness from its predecessor, which was the number of times the path changed from black to white or white to black.

The second run was designed to find a set of descriptive statistics for each of the letters in the frame individually. First, a transparent overlay was put on to each of the line printer pictures resulting from the first run, and shuffled to find the displacement of the frame from the assumed position. This was necessary, as each target had been put on to the oscilloscope face by hand, without registration marks, so that each would be in a slightly different position. The information about this displacement, about the letters in each frame, and about the target letter, along with enough information to identify the frame, were punched on to paper-tape. This tape, and the data types, were processed during a second run to produce the same set of statistics for the area surrounding each letter.

A set of statistical routines converted this information into a set of results in readable form.

Chapter 4

PILOT EXPERIMENTS

Experiment 1. To Test the Effect of Different Pictures

Introduction

The pictures used for these experiments have several constraints on them. The first is that the detail in the picture must be bigger than the resolution of the scanner, about 1 mm. Finer detail will now show up reliably. This rules out line drawings. They must also be small enough to fit on the scanner. (75 mm sq.)

Among the other possibilities those tested are:

1. Road signs, which, although they contain some fine detail, are designed to be recognisable without it.
2. Specially drawn shapes, designed to be different, but constructed of blocks of black. I am grateful to William Edmondson for these.
3. Letters whose bars are larger than the resolution of the scanner.

The experiment was designed to establish the advantages and disadvantages of each type of material.

Informal trials had suggested that an uninstructed subject tended to move the pen as fast as possible. However, the aims of the investigation required that the subject moved slowly, and with consideration. Two possible instructions to the subject, in order to achieve this, were a minimum distance condition, in which the subject was asked to move the pen as little as possible, and a minimum crossingscondition, in which the subject was asked to minimise the number of times the pen crossed an edge in the picture. This last was intended to encourage the subject to move the pen within the figure. Informal trials had also suggested that single letters were easy to recognise, and that recognition was little disturbed by putting the letters against a background, or by distorting them.

Apparatus

The X-Y plotter, joystick, diagonal scan generator and flying spot scanner were used. Since the A-D converter and punch were not yet available for measuring the distance moved during the trial, a second flying spot scanner with a mask of random dots was coupled to the system. The edge crossings were counted, giving a measure of distance moved, which had no preferred direction of motion. This method introduces a stochastic error of the order of the square root of the number of counts.

The targets used are shown in Figures 4.1, 4.2, 4.3.

Subjects

The eleven subjects tested were fellow students in the age range 18-25. Seven were male, four female. They were aware of the general purpose of the experiment, ie, an investigation of pattern recognition processes.

Procedure

The subject was asked to identify the letter or shape presented. He was given lists showing the road signs and Edmondson figures. Some trials were controlled by joystick (ie, manual), others by the diagonal scan generator.

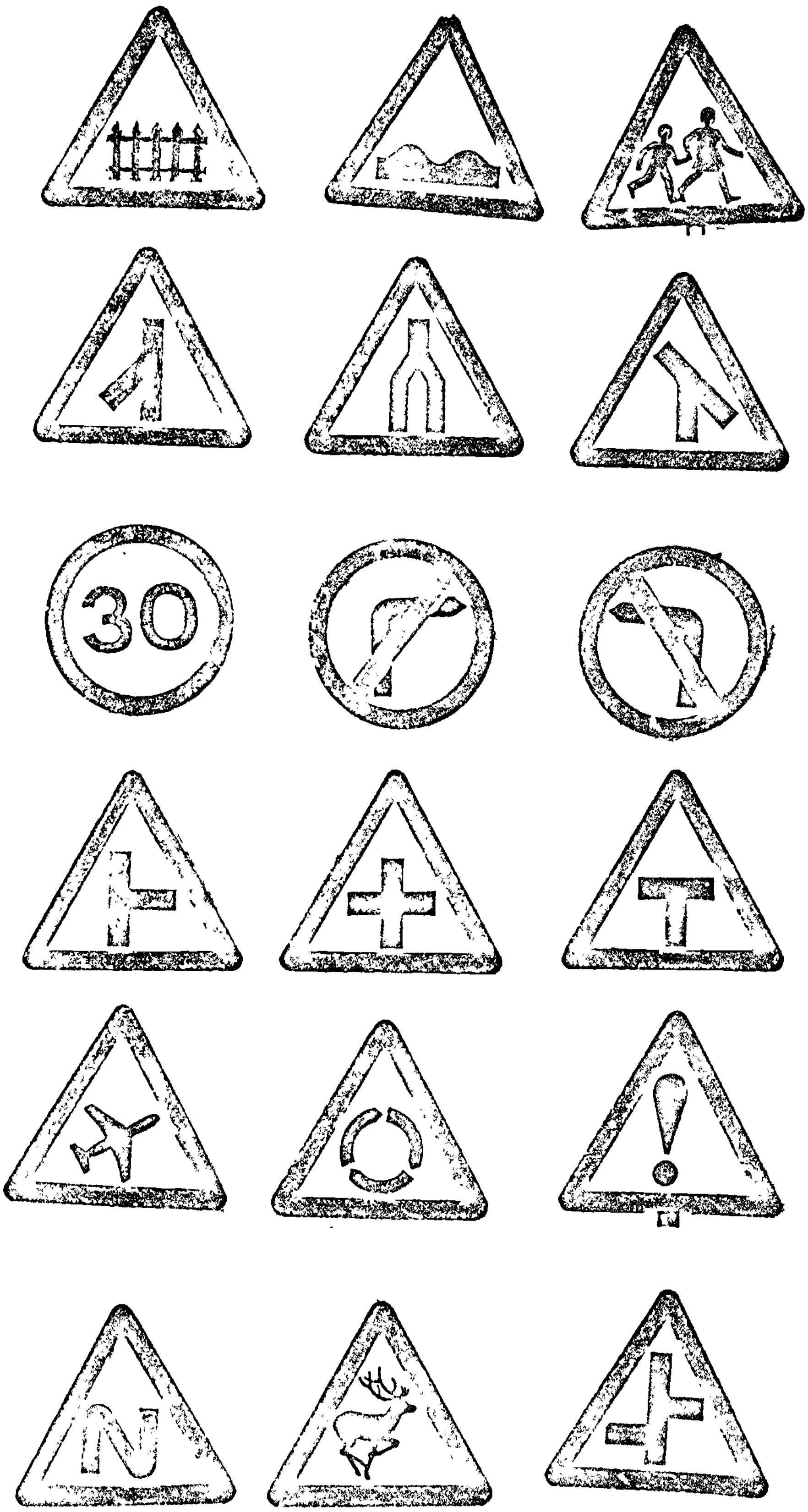


Figure 4.1

Road Signs

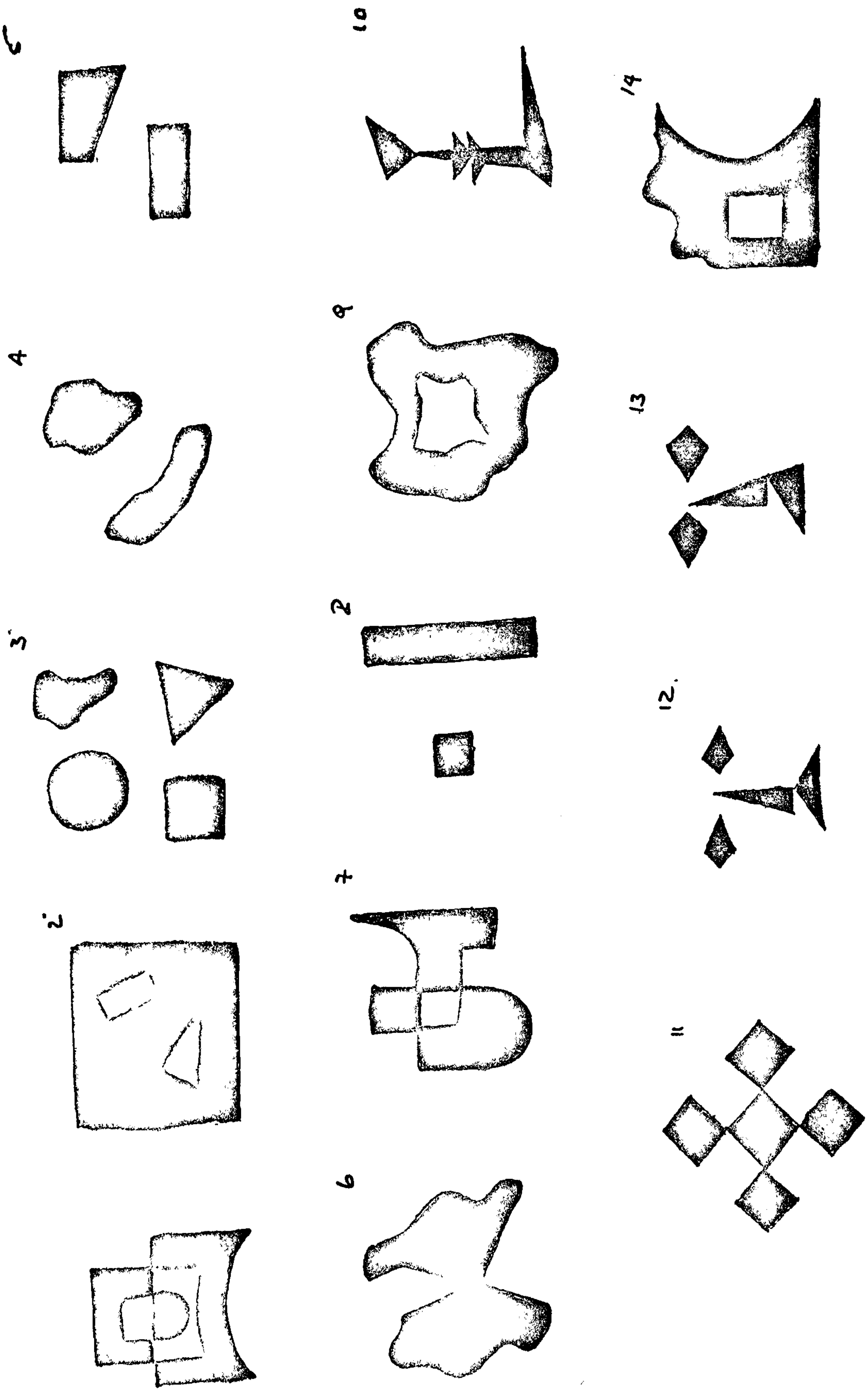
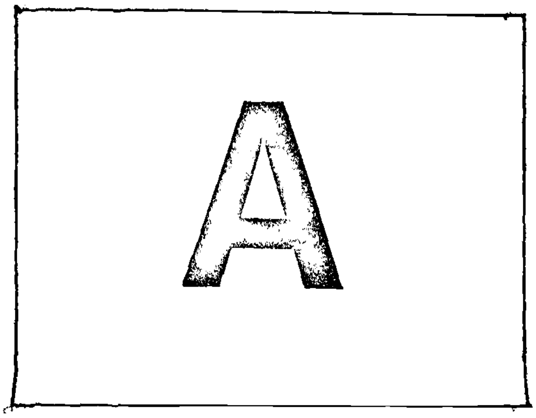


Figure 4.2

Edmondson Shapes



Layout in frame

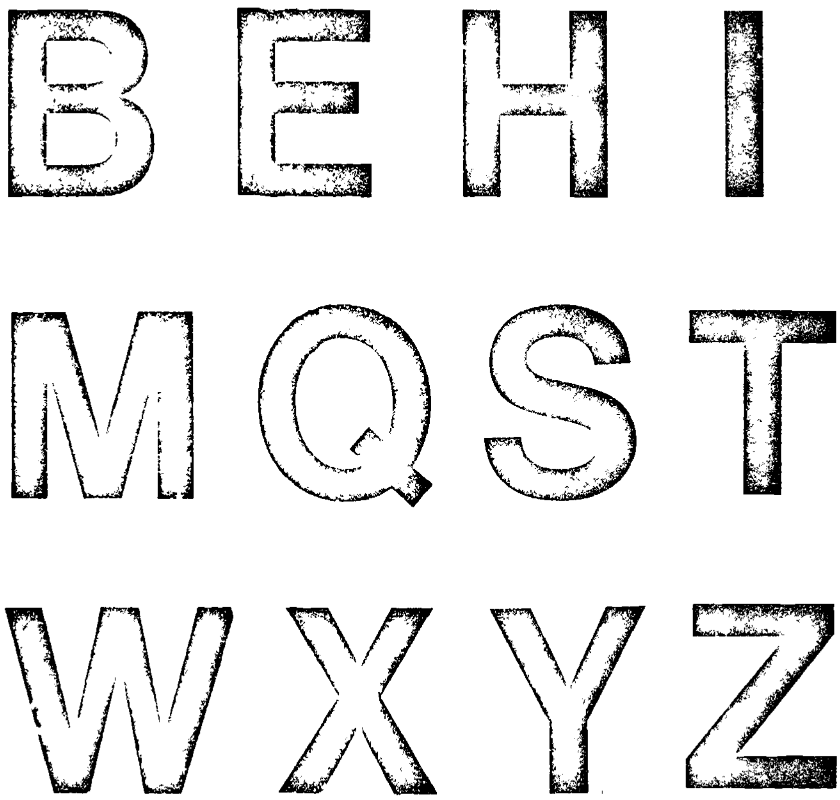


Figure 4.3
Letters used as targets

Twenty trials were presented in the order:

No of trials	Patterns	Scan	Subject asked to minimise
5	Letters	Joystick	distance
3	Letters	Machine diagonal	
3	Letters	Joystick	crossings
3	Road signs	Joystick	distance
3	Shapes	Joystick	distance
3	Shapes	Joystick	crossings

Results

The results are shown in Table 4.1. They are reported as Median and Interquartile intervals because there is a numerically small tail of very large results. The means and standard deviations were more affected by these extremes than seemed warranted, and the standard deviations were often larger than the mean, which, for quantities which are necessarily positive, indicates that statistics implying normal distributions are not appropriate. It was felt that reporting the median and interquartile intervals was a more veridical representation of the data.

Most subjects could recognise the stimuli after they had understood what was being asked of them. Several subjects required an explanation from several viewpoints before they understood. The most effective explanation of the general task, and the one that was afterwards adopted, was as follows:

1. Show the subject a very densely mechanically scanned image which he could easily recognise;
2. Explain how that was drawn;
3. Substitute another transparency in the scanner, and allow

	Distance to Recognition (Arbitrary Units)		
	Median	25%	75%
Manual Scans			
Letters			
Minimum distance	107	71	152
Minimum crossings	118	85	194
Road Signs			
Minimum distance	172	130	260
Edmondson Shapes			
Minimum distance	99	73	197
Minimum crossings	135	80	265
Diagonal Machine Scan			
Letters	134	109	251

Table 4.1

Results of first pilot experiment

the subject to experiment.
Neither of the transparencies shown in this way was presented to that subject again.

The road signs were unsatisfactory as targets because many of their discriminating features were too fine to be resolved by the apparatus. Many subjects found it difficult to distinguish surround from interior and, rather than ignore the surround, filled it in completely before going on to the centre. Also the rhomboid distortion introduced by the scanner made recognition difficult.

The Edmondson shapes were more satisfactory targets. Because of their diversity more early recognitions were possible than was the case for road signs. The letters were also easily recognised, but no more efficiently than the patterns.

Many subjects at first treated the task as if they had been asked to recognise the target in the minimum time possible. These subjects were verbally restrained, since the pen, being rate limited, did not follow the subjects' movements accurately. Also, another error was introduced: since the pen took a finite time (25 msec) to fall from its raised position to the paper, crossings of the same edge at high speed in different directions indicated the edge to be in different places. At slow speeds this effect was negligible.

Under joystick control the difference between trials under the minimum distance instructions and those under minimum crossing instructions is in the expected direction, ie, those under minimum distance instructions are shorter, but the difference is small and not statistically significant. $p > 5\%$ (median test). Most subjects appeared to treat both sets of instructions similarly.

The diagonal machine scan produced distances to recognition that were in general longer than those for manual scans under either type of instruction but this was not significant (median test): however, it prompted further investigation.

Discussion

The similarity of the distance to recognition of letters and of Edmondson patterns was unexpected. Subjects scanning an Edmondson shape had to compare this with a list in front of them encouraging them to scan the whole pattern, and then stop and match it to the samples provided; whereas letters are very familiar and subjects scanning a letter may be presumed to have an internal representation to which they can easily refer.

This finding is compatible both with the results of the later experiment with shapes and the more general theory advanced from the evidence of all the experiments. We shall therefore refer to it again in the summary of the experiments.

Experiment 2. To Test the Effect of Different Scans

Introduction

Machine scanned trials had been introduced in the previous experiment because preliminary trials had shown that the subject scanning manually moved the pen so fast that the outlines became blurred. This could be avoided by using a regular, slow scan, produced by its own generator. The diagonal scans were apparently less efficient than normal. Several other machine scans were made to determine whether this inefficiency was due to uniform scanning or to the effects of diagonal lines. These scans were arranged to be analogues of the eye movement patterns found by Howe (1965), and are illustrated in Figures 3.10, 3.12, 3.14, 3.16.

Procedure

Twelve students, six male and six female, served as subjects in this experiment; their age range was 18-25. The subjects were told that it was an experiment in pattern recognition. The transparencies used were the single letters and the Edmondson shapes shown in Figures 4.2 and 4.3. The subject's task was to recognise the letter or to find in a list the Edmondson shape present.

The order of presentation was:

Order A

Number of frames	2	2	2	2	2	2	2	2	2	2	2	4
Scan type	@	⊡	≡	|||	∝	✋	@	⊡	≡	|||	∝	✋
Transparency type	L	L	L	L	L	L	S	S	S	S	S	S

Order B

Number of frames	2	2	2	2	2	2	2	2	2	2	2	4
Scan type	∝	|||	≡	⊡	@	✋	∝	|||	≡	⊡	@	✋
Transparency type	L	L	L	L	L	L	S	S	S	S	S	S

6 subjects used Order A, 6 Order B.

L = Letter

S = Edmonson Shape

@ = Round spiral machine scan

⊡ = Square spiral machine scan

≡ = TV(X) machine scan

||| = TV(Y) machine scan

∝ = diagonal machine scan

✋ = manual scan

In the previous experiment the subjects had responded in the same way to the instructions given to minimise distance travelled or crossings. In this and subsequent experiments subjects were asked to minimise the total distance moved before recognition. The distance was measured in the same way as in the previous experiment.

<u>Results</u>

The results are shown in Table 4.2. The new scans, with the exception of the round spiral are more efficient than the diagonal scan. The best, the TV(X) is very similar to the manual

Scan	Distance to Recognition (Arbitrary Units)		
	Median	25%	75%
Letters			
	81	52	125
	75	55	128
	100	72	138
	94	78	160
	145	114	207
	135	77	221
Edmondson Shapes			
	104	70	176
	105	78	129
	128	67	162
	111	73	142
	130	100	212
	115	95	220

Table 4.2

Results of Second Pilot Experiment

scan. The only significant difference is that between the TV(X) and the diagonal machine scans.

Discussion

A surprising aspect of the results is their small spread. The difference between scans is similar to the range within one scan. This scatter within each scan is only partly due to individual differences between subjects. Some subjects found the task very difficult, and their results constituted a high tail. However, for most subjects the results spread over most of the range, rather than being concentrated at one value. Thus the variability is due to the task itself, and the measurement in it, rather than to differences between subjects.

The inefficiency of the diagonal scan relative to other machine scans is unexpected, as the scan path covers the plane as uniformly as that of TV scans. It may be that subjects are biassed to accept information in rectilinear form, not diagonal. Loomis (1974) found that subjects could recognise letters presented tactilely more easily when they were scanned with a vertical slit than a diagonal, which suggests the same bias in his subjects.

General Discussion

The introduction argued that scanning is a useful way of investigating recognition. These experiments showed that the apparatus and technique elicited scanning which could be studied in this way. Subjects were able to understand the task after the explanation had been improved, could readily use the controls and equipment, and found the task moderately, but not impossibly, difficult.

Of the three sets of patterns investigated, road signs gave difficulty, while Edmondson shapes and letters gave similar results. It was decided to concentrate on letters, because these were well-known.

The results of using the machine scans showed little difference between manual and ≡, |||, scans. was apparently less efficient and significantly so. The three rectilinear scans were used as comparison scans in the next experiment, as being the most efficient of the uniform scans. Though the subject could be restrained from working at high speed by an instruction to minimise some quantity, whether this quantity was distance or crossings made no apparent difference to his scanning behaviour. In subsequent experiments the subject was asked to minimise distance.

As the expected error in measured distance moved was similar in the pilot experiments to the differences caused by different types of scan, it was felt that a more accurate measuring system was required.

The subjects scanning rarely appeared to be the most efficient that could be thought of and usually covered the whole letter. The subject often appeared to continue scanning a letter after giving some indication that he knew what it was.

Some subjects used hypothesis testing, both by the experimenter's observation and their own report.

These findings are compatible with the results of later experiments, and will be explained then.

The pilot experiments validated the technique; the paradigm experiment explores the task it was designed for, testing feature analysis and hypothesis testing models of recognition.

Chapter 5

THE PARADIGM EXPERIMENT

Introduction

This experiment was designed with a task best solved by hypothesis testing, ie, if subjects can use hypothesis testing, they should do so here. If the subject solves the task some other way, we are led to deduce that hypothesis testing is not available to him.

As has been explained in Chapter 2, hypothesis testing will be best elicited by a situation where a subject has to scan a display to find a known target. If the scan is permanent, then it can be assessed later for the characteristics of feature analysis and hypothesis testing.

In the pilot experiment, the subject had been asked to recognise both letters and shapes. It was observed that some difficulty was caused in the recognition of the shapes because of the need to search the sheet showing all the shapes before deciding which one matched the delineation so far revealed.

Because of the considerations, explained in Chapter 2, that the recognition of shapes is ambiguous without a specimen sheet, and that searching a specimen sheet for shapes confuses the subject and the interpretation of results, letters were used as patterns for this experiment.

The task is to find a target letter, which is held constant throughout the experiment, among other letters. This allows a subject time and experience to produce a specialised scan technique for that letter only (as is done for the hypothesis testing simulation).

As a comparison, machine scans, which scan the whole field in straight lines, and thus produce the easiest delineations to feature analyse, are used on 3/5 of the trials. The scans used are ≡ & |||, for simple uniformity, and , for uniformity including corners. The diagonal and circular machine scans were omitted to keep the number of trials down.

The logic of this study demands a matching task, rather than an identification task, as was explained above. A plurality of letters in each frame allows the subject to report the letter most like, or least unlike the target, by comparing the partial delineations available. However, as there are more letters in each frame, the possibilities of confusion grow. We have already seen that one of the difficulties is that of relating the scan segments. As the density of letters increases, scan segments from two letters become more likely to be confused.

Because the implications are unclear, the experiment was set up with sets from 1 to 5 letters in each frame.

Apparatus

The A-D converter and the analysis system described in Chapter 3 were used in this and subsequent experiments.

Targets

For the third experiment, two series each of 25 transparencies were made. These are shown in Figures 5.1 and 5.2. These were in batches of 5, each with the same number of letters per frame. Each batch had a target letter which was included in every frame, and other letters, to make up the required number, were selected by random numbers. The letters selected, including the target letter, were then allocated by random numbers to the possible

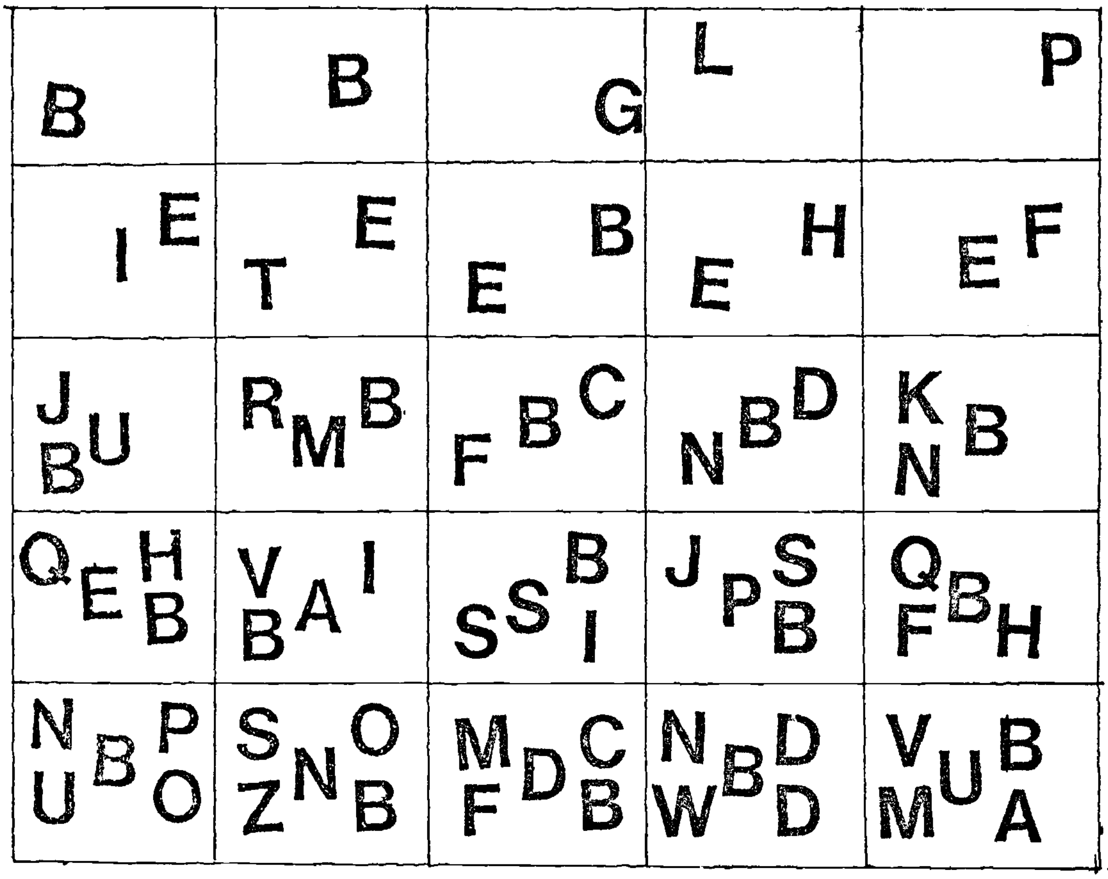


Figure 5.1
Target letter normally "B"

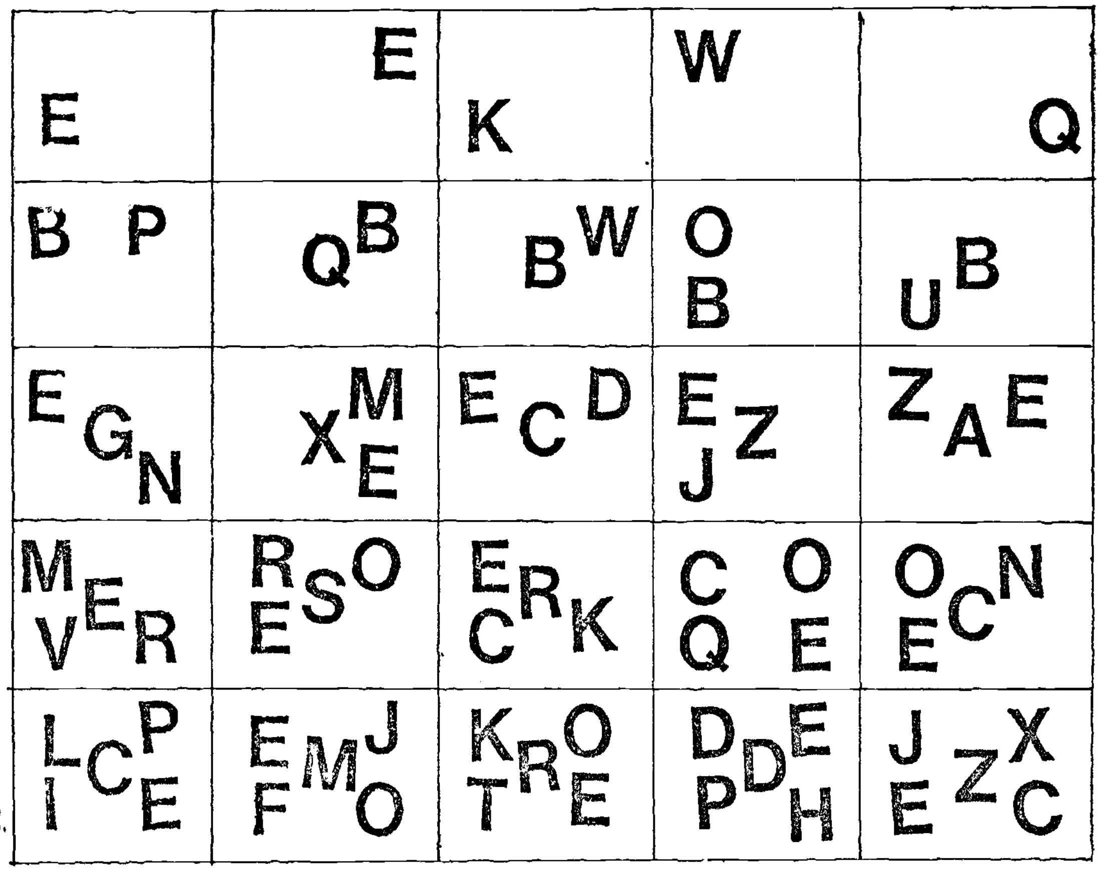


Figure 5.2
Target letter normally "E"

positions in the frame.

The single letter batches, for which this procedure was in-applicable, were generated by taking two target letters and three randomly chosen other letters, and putting one in each frame in a randomly chosen position.

The choice of target letters, also made by random number selection, was "E" and "B". These were unfortunate choices, in that they are both visually and aurally similar. When scanned from top to bottom, both produce three short strokes and three "clonks" of the pen lift mechanism, which is a recognisable signature for these letters, and only these letters.

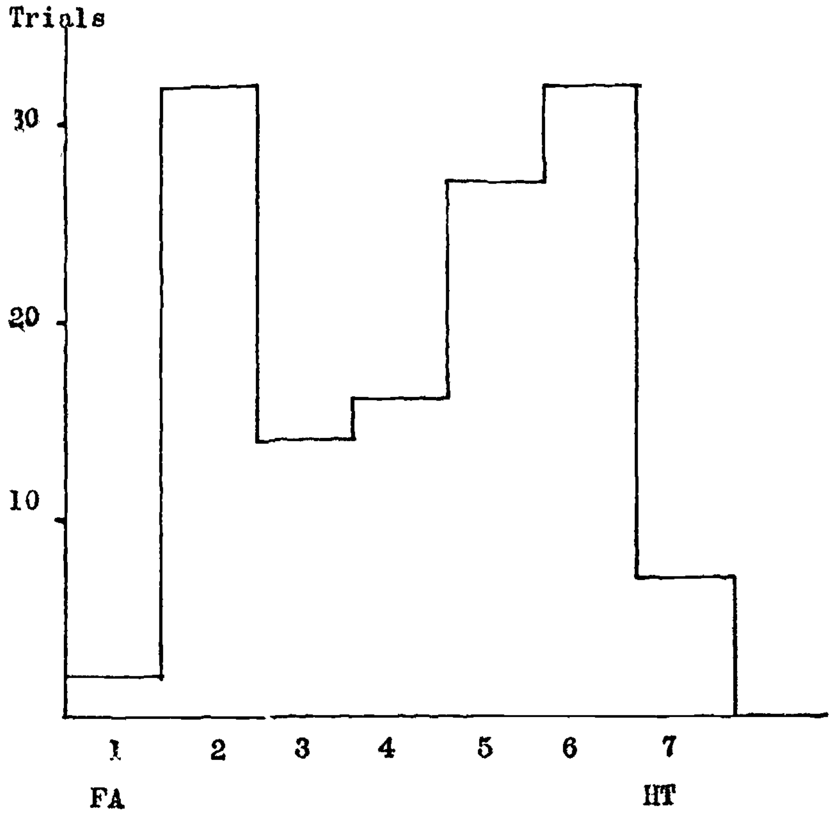
The transparencies were made by sticking 25mm 'Letrasign' letters, in Helvetica Medium typestyle, sans seriph, to clear acetate film.

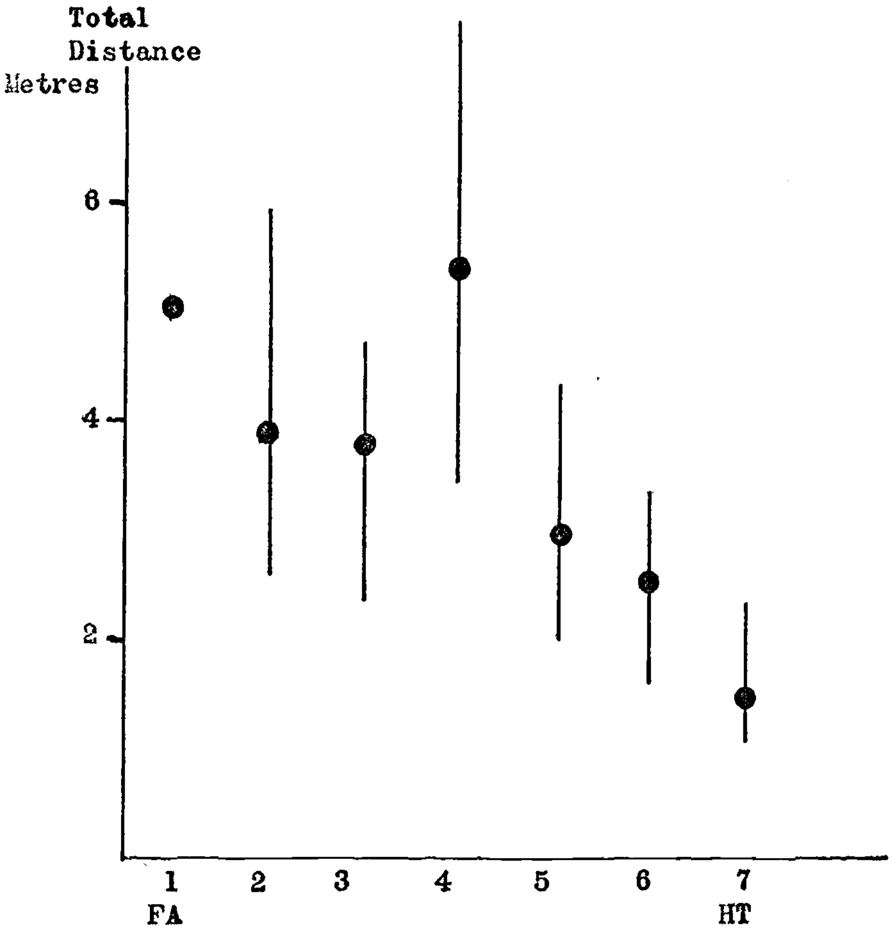
Procedure

Trials were arranged in batches of five, all trials in a batch with the same target letter and number of letters per frame. Subjects were informed about the number of letters in the frame and of the target letter whose position they were asked to report. When only one letter per frame was presented the subject was asked to report whether or not it was the target letter.

Within each batch of five trials the scans,  ,  ,  ,  ,  were randomly ordered by card shuffling. The order of presentation of targets was also randomly ordered by shuffling. On manual () scans the subject was asked to move the pen through the minimum distance possible.

Two sets of subjects were tested. Ten subjects, second





data are reported as median and interquartile intervals

Figure 5.3
Distribution of assessed Feature Analysis-Hypothesis Testing

Figure 5.4
Total Distance against assessed Feature Analysis-Hypothesis Testing

year psychology students, seven male and three female, age range 18-23, were tested using the targets shown in Figure 5.1. Nine subjects, students from several faculties, all male, age range 19-25, were tested using the targets shown in Figure 5.2. Subjects were paid 5/- for one session of about an hour.

Trials were terminated when the subject reported, or after 5 minutes. The experimental session was terminated at the end of the batch of 5 trials, after an hour.

Results

This experiment was designed to find whether subjects used a continuous correction (hypothesis-testing) mode of scanning. The delineations of target letters made in the experiment were judged, by the experimenter, on a seven point scale, for the degree to which they showed the characteristics of hypothesis testing (small movements, oddly distributed over the letter) or of feature analysis (uniform scans over the whole letter). The results are shown graphically in Figure 5.3, and indicate that there was a spectrum of behaviours, most being neither pure feature analysis nor pure hypothesis testing. These are judgements made of single trials, not averages over subjects; therefore the intermediate values are not due to a subject oscillating between distinct strategies on different trials. The most common cause of an intermediate rating, 3-4-5, was that the subject concentrated different area-covering scans on different parts of a letter. One common occurrence, arguing for oscillation within a frame, was that the subject scanned in some way over parts of the letter, usually the target letter, and then attempted to draw a line along the letter bars, which could be seen as a final stage of hypothesis confirmation, usually following a basically feature analysis recognition phase.

The judgements of feature analysis and hypothesis testing are not absolute, but made so that most are on the middle of the scale. The distribution of judgements does not differ significantly from a hypothetical distribution with none judged 1 and equal numbers judged 2-7. ($p > 75\%$, Kolmogorov-Smirnov test.)

In absolute terms, only category 7 is unambiguously hypothesis testing and 7 out of 128 trials were put into this fully hypothesis testing category. Nevertheless, trials assessed high on hypothesis testing took less distance to recognition than those low on hypothesis testing. (Median test, $p < 1\%$.) The relationship of distance to hypothesis testing judgement is shown in Figure 5.4.

Overall in the manual trials the common pattern was a general scribble over the area occupied by a letter, followed by a scan along its lines, perhaps a confirmation of a judgement already made, and hypothesis testing only in a weak sense.

This pattern of action nevertheless leaves a trace which is similar to one using hypothesis testing at an earlier stage, in its weaker form of guessing where the bars of the letter will be and then moving along them. Both forms will be judged intermediate in hypothesis testing, because they cannot be discriminated from what is on the paper.

No subjects used an edge following strategy, and few a bar-following strategy. A bar-following scan, moving the pen across the direction of the bar found so far, would appear to be an economical scan. Most scans were across or up and down, with a few in a diagonal arc which was, I suspect, comfortable to move in.

Considering distance to recognition for the whole frame, the machine scans required more scanning to render the target letter recognisable than did the manual scan. The order of machine scans (Figure 5.5) was ⊡ ≡ ||| , although the differences were not significant. (Median test, $p > 10\%$.) The manual scan took about 60% of the machine distance to render the letter recognisable.

The pattern is very different when the results are analysed as the distance per letter rather than for the whole frame. This was done by calculating the same statistics (as for the whole frame) for parts of the scan curve lying within a rectangle surrounding the letter. (Figure 5.6.) These values will be referred to as distance per letter, crossings per letter, etc.

Plotting distance per letter, against scans, (Figure 5.7), the order is nearly reversed ||| ≡ ⊡. The change in relative position of the ⊡ , relative to the ≡ . ||| scans, between distance/frame and distance/letter, is because the ⊡ covers a slightly smaller area than ≡ ||| scans, and because by its nature it is less densely scanned at the edge, and so scanned less blank space outside the letters.

The difference in the relative positions of the manual scan is because the subjects scanned letters and ignored blank spaces, while the machine scanned uniformly everywhere.

Letters were assessed for legibility (Table 5.1). For manual scans 95% of target letters were legible and 58% of non-target letters, implying that subjects concentrated on target letters. For machine scans, target letters were not significantly

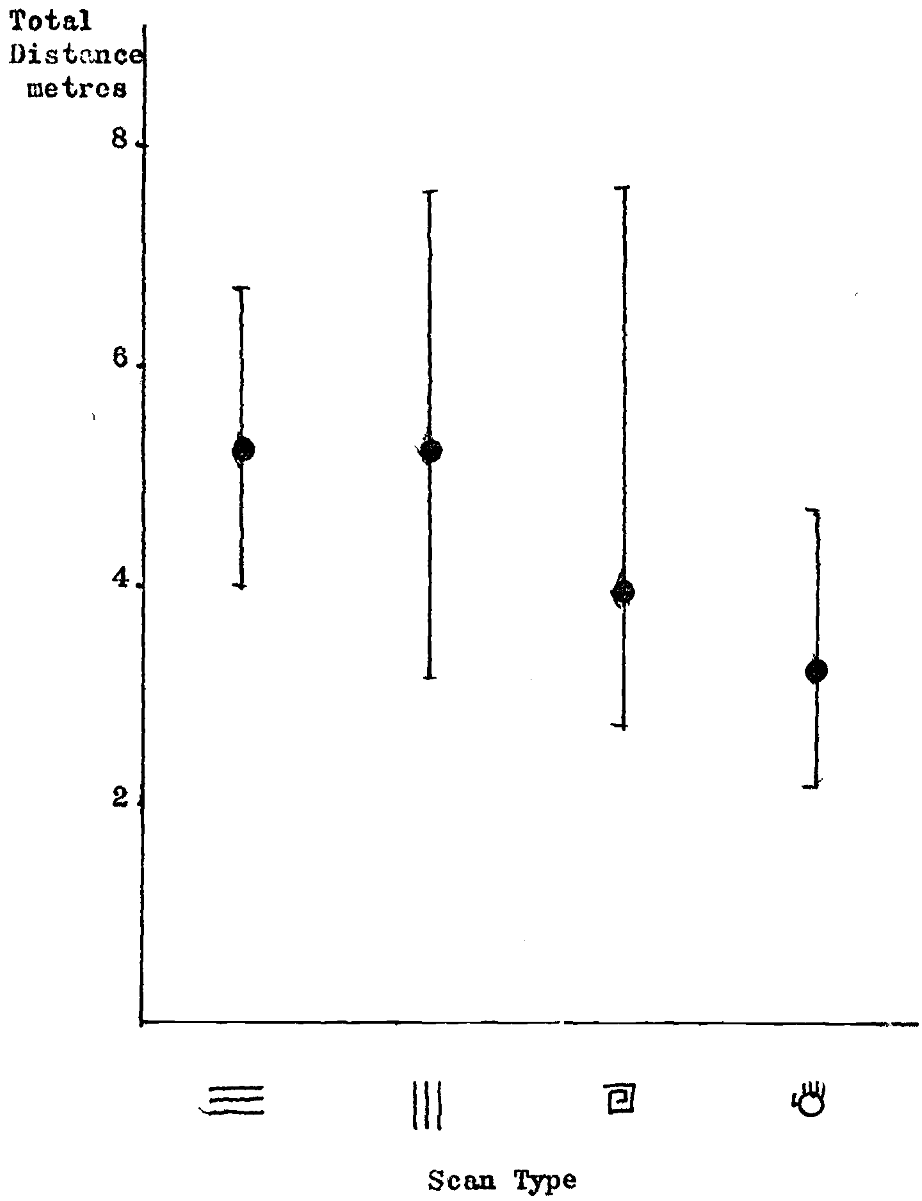
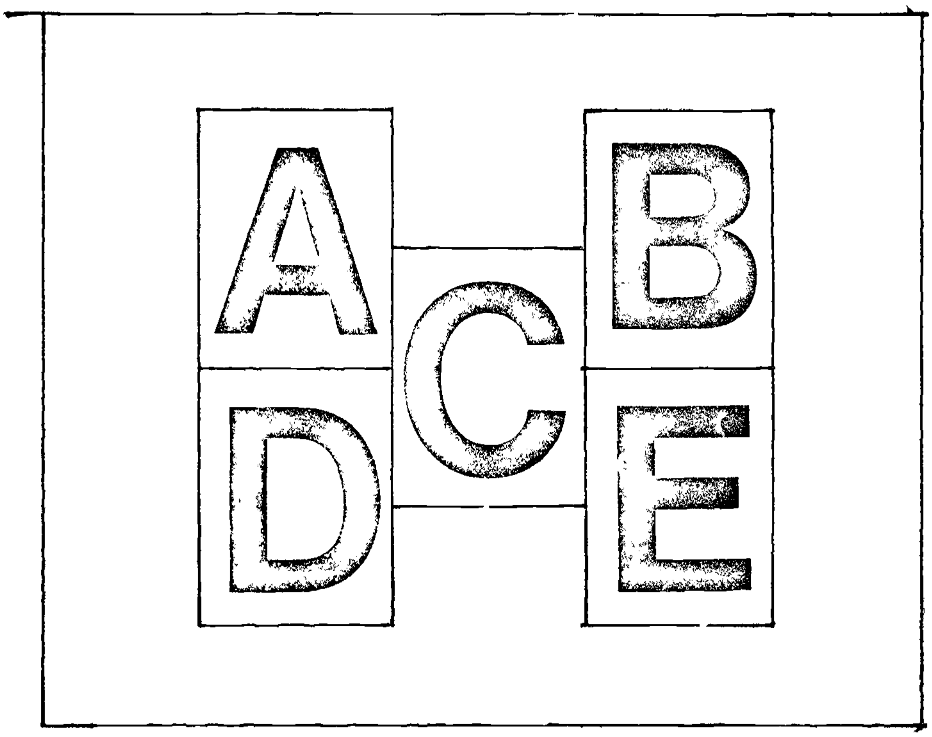
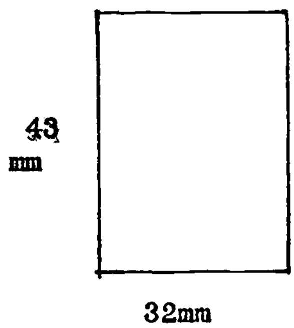


Figure 5.5
Total Distance against Scan Type



Layout of frame, showing cells surrounding
each letter



normal cell size (on the plotter)

Figure 5.6
Letters per frame - explanation

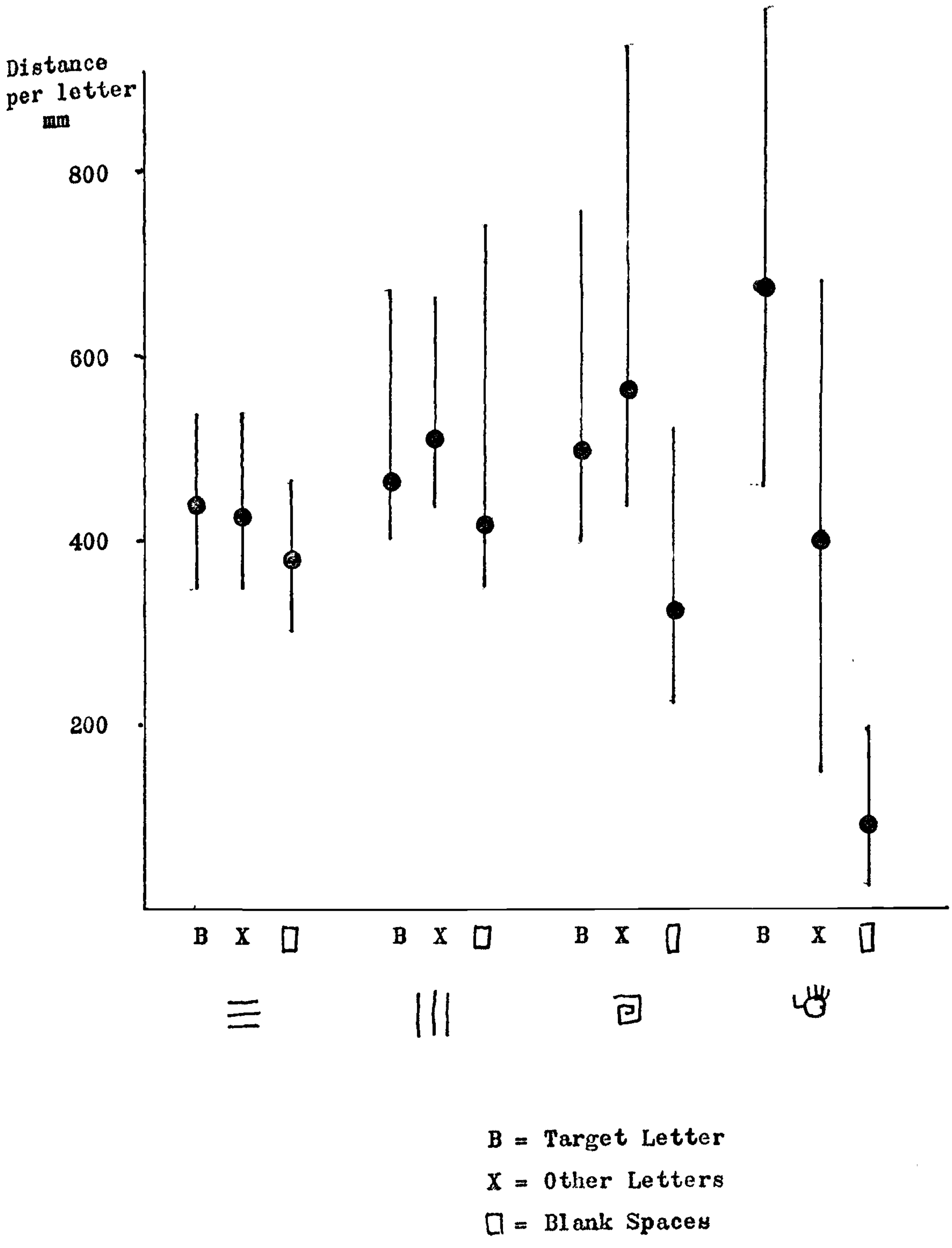


Figure 5.7
Distance per letter against scan type

Each delineation of a letter was assessed as being

1) readable- immediately
2) deducible - with thought

   these two categories are combined as legible.

3) illegible - at all.

Scan	Target Letters			Non-target Letters		
	Readable	Deducible	Illegible	Readable	Deducible	Illegible
	80	9	11	82	9	9
	59	20	21	66	8	26
	51	40	9	68	23	9
	83	12	5	35	23	45

Percentages

Table 5.1

Legibility of letter delineations

differently legible from non-target letters, whereas they were for manual scans. (x^2 test $p > 0.05$ or $p < 0.001$.) No difference would be expected from uniform scans.

The delineations of target letters were assessed for coverage and density. The median coverage of letters in all scans was 80%, and only 6% of manually scanned target letters were less than 50% covered, arguing that effectively the whole letter had to be visible for recognition. The median density of target letters on an arbitrary scale was 3 for ≡ , 4 for ||| (and 5 for &), 3 was the lowest density at which the separation of scan lines is narrower than the letter bars. This is in agreement with the predictions of feature analysis.

There is a very prominent difference between subjects' distances to recognition. (Median test $x^2 = 45.4$, 16df.) This is still significant, though much reduced, if the two most hypothesis testing subjects were excluded. ($x^2 = 27.2$, 14df.) In other words, two subjects were very different from the rest, in that they used hypothesis testing and were very efficient.

Considering the number of letters per frame and distance per frame, the results are shown in Figure 5.8. The manual scans increase by about 200mm/letter, the ≡ by 430mm/letter, the ||| by 290mm/letter, and the result of the is not apparently a straight line. The manual scan value is low, about half the median distance per letter for non-target letters, as would be expected for subjects scanning letters one by one until finding the target letter.

The increases in machine scans are greater, and are presumably due to the increase in possible confusions with more

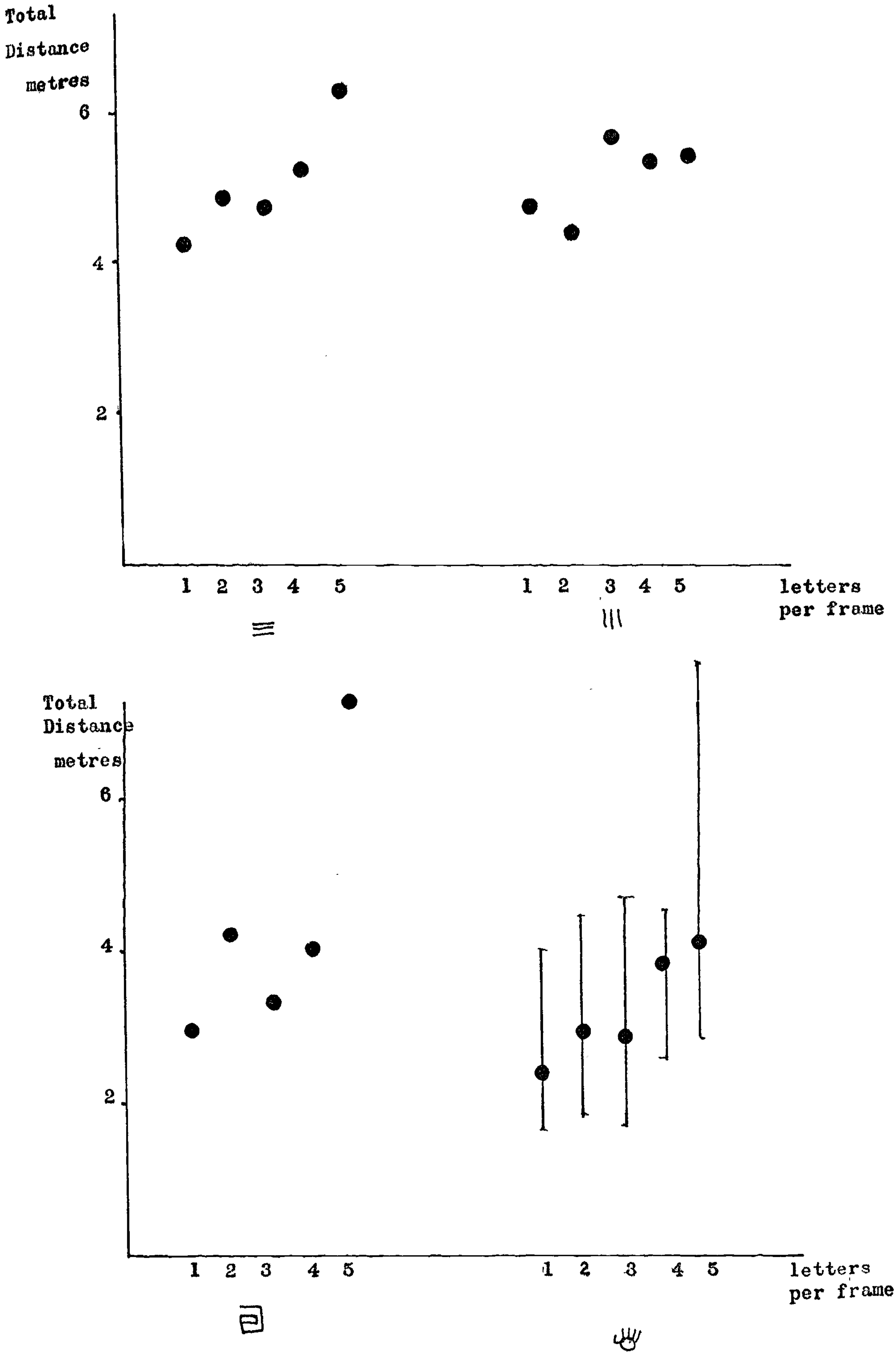


Figure 5.8

Letters per frame against distance

letters in each frame. Subjects' most popular manual scan type is vertical, which is the one here giving fewest confusions, possibly because the height of a letter is fixed for the whole alphabet while the width varies. The difference made by changing the number of letters per frame is small.

The median of the average pen speed for each trial with manual scans (4.9 mm/sec) is between that for (6.3 mm/sec) and (2.9 mm/sec). The curve of velocities against assessed hypothesis testing is shown in Figure 5.9. Subjects appeared to scan fast normally, then stop and look, which reduced their average pen speed. In manual trials, subjects nearly all scanned letters one by one, passing to another when they had identified the first. This impression comes from observation of their behaviour, and is corroborated by the distribution of legible and part-legible letters, and by the distributions of distance/letter, where the distance for target letters is much greater for target letters than for other letters, and virtually nil for blank spaces. With this hypothesis, one would expect the distribution of distances per letter for non-target letters to be bi-modal - those where recognition has been attempted being as much scanned as target letters, and those where no recognition has been attempted looking like blanks.

The result, shown in Figure 5.10, is compatible with a combination of values near zero added to a set with the same distribution as target letters, but is not unequivocally bi-modal.

Conclusion

Most of the evidence suggests that subjects use a feature analysis form of recognition.

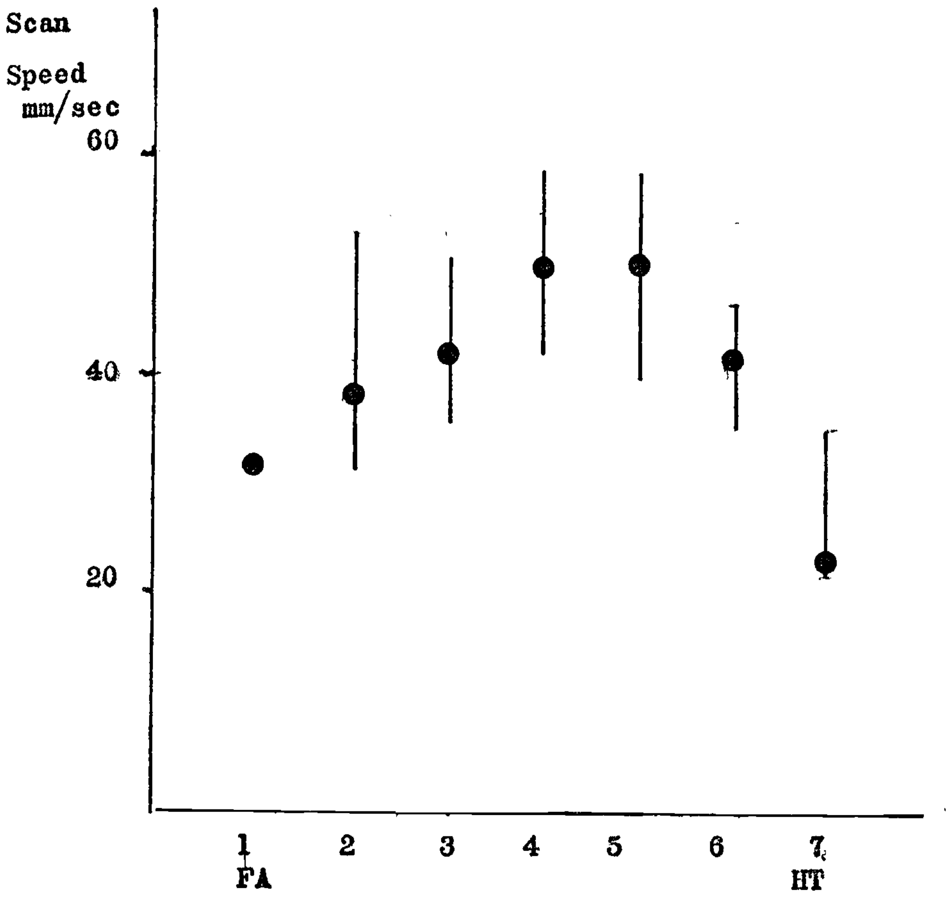
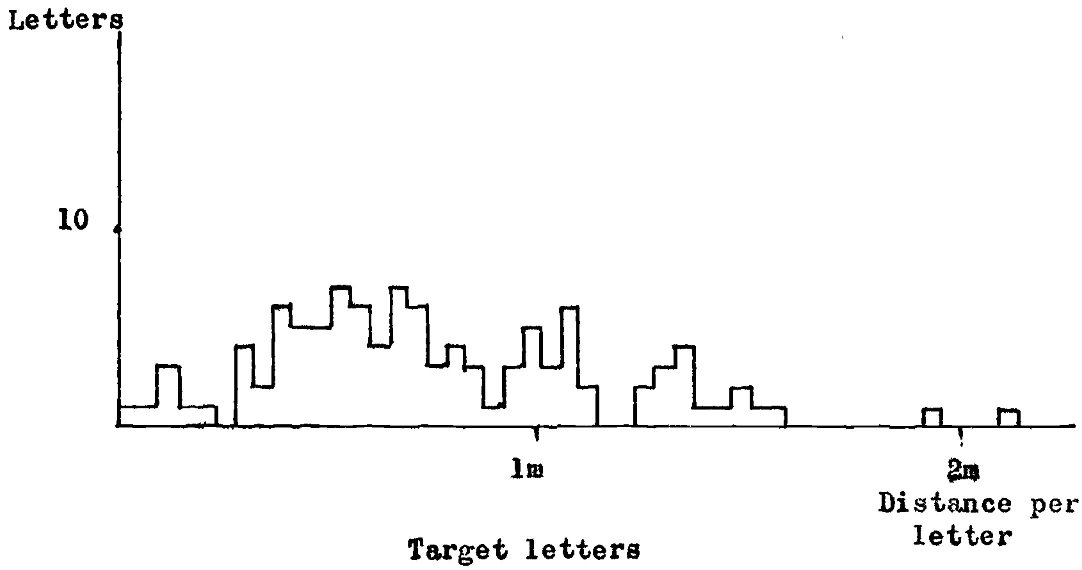


Figure 5.9

Scan speeds against assessed Feature Analysis-Hypothesis Testing



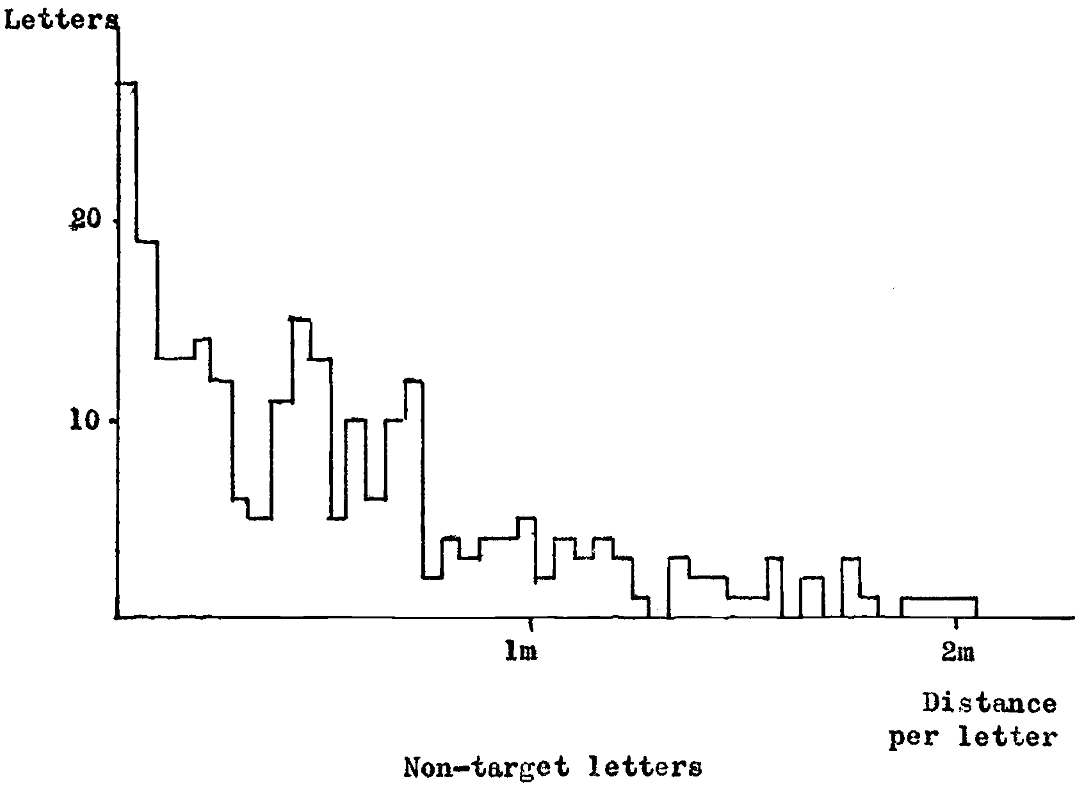


Figure 5.10

Distance per letter for target and non-target letters

First, only 7 trials out of 128 were unambiguously hypothesis testing. These were successful, which was expected, and it is surprising that hypothesis testing was so rarely used. Other trials were judged near hypothesis testing, but the absolute meaning of this must be found from other evidence.

Second, the machine scans needed less distance per letter to make one letter recognisable. This is only likely if recognition is by feature analysis.

Third, changing the scan method, the number of letters per frame, or the subject's adopted scan strategy does not change the distance to recognition by more than a factor of two. The only exception was among manual scripts, where those judged very high on hypothesis testing were less than half the overall median distance. Hypothesis testing engenders manual scans much more efficient than machine scans, and so hypothesis testing seems intuitively likely to lead to greater variability than feature analysis.

Fourth, subjects scan essentially the whole letter, and the resulting script is legible to the experimenter.

However, when scanning manually subjects concentrate on one letter at a time; this need not mean that their scanning is influenced by the <u>form</u> delineated, but is influenced by the existence of some successful scan lines on the page, which indicate the position of a letter.

Manual scans took the most distance per target letter and the least distance overall, explicable only by the subject ignoring blank spaces. Also, the subject ignores some letters. The evidence for this is that target letters are much more likely to be legible from the script than non-target letters, and many letters had no trace in the manual conditions. This never happened in machine trials. This is consistent with subjects scanning and recognising one letter at a time until they find the target.

The subjects scanning is much more dense than it need be, as is shown by the greater density of manual delineations than machine ones. The original conception of feature analysis had a rule-governed scan giving a regular scan pattern. An alternative to this rule is to scan next to a line already revealed. This will avoid scanning in blank spaces, but is more likely to scan nearer the previous line than the width of a letter bar, because the correlation of parallel scan lines one letter bar width from each other is only moderate. These predictions are consistent with the observed behaviour of the subjects. The rule does not require partial recognition of the letter for its execution (and this is compatible with feature analysis) but does require feedback from the page.

So, the evidence here is for feature analysis as the overall recognition method, but that subjects control their scanning to where there is already scanning, which would not be expected under pure feature analysis.

Chapter 6

CONSTRAINED MANUAL SCANS EXPERIMENT

Introduction

The previous experiment had shown that subjects used feature analysis, and that the most efficient way of delineating a letter was to scan it with a uniform machine scan.

It is also possible that hypothesis testing failed to show itself before because the lines drawn by manual scanning are rarely straight and thus, for example, the blank parts of the scan convey no information, because the line across them cannot be mentally reconstructed. If straightness is an advantage then a scan system with human placement of straight lines would show hypothesis testing, and be the most efficient form of scan, as it could combine the reconstructability of straight lines with the human ability to place a line where there is something useful to be revealed, and avoid scanning blank spaces.

Thus if we give the subject a control system that easily draws vertical and horizontal lines, which will enable him to reconstruct blank traces, he may be enabled to use hypothesis testing.

If this equipment shows that subjects can use the two rectilinear joysticks to operate a hypothesis testing recognition scan, then the lack of hypothesis testing in the paradigm experiment must be an artefact from the flexibility of control by the single joystick.

Also, in the paradigm experiment the parameters of the machine scans (separation of lines and turning points) were chosen arbitrarily. If these parameters are controlled by the subject, this should give the most efficient form of scan, with the advantages both of regular scanning and of optimum placement of lines. This could happen with either feature analysis or hypothesis testing. Both these aims may conveniently be achieved by giving the subject a joystick with separate horizontal and vertical controls.

Procedure

To facilitate comparisons the task given was the same as for the previous experiment; to report the position of the target letter after the least possible movement. Those targets shown in Figures 5.1 and 5.2, whose target letter was "B" were used throughout. The procedure for the selection of targets and scan was the same as before, both randomised by card shuffling. Wherever a machine scan was called for by this shuffling, the subject was told to produce this scan pattern with his own spacing, using a pair of joysticks controlling the two dimensions of movement independently. In the TV(X) manual scan (symbol ↔) he was asked to scan from side to side right across the picture, but at whatever height he wanted, and similarly for the TV(Y) manual scans (symbol ↕). For the Square Spiral manual scan (symbol □) he was asked to move left, up, right, down, in that order, but for any distance he chose. The manual scan trials were conducted with the same instructions and the same joystick as in the paradigm experiment, interleaved with the constrained scans.

Nine second year psychology students, age range 18-22, two male and seven female, served as subjects. None had served as subject for a previous experiment. They were paid 5/- for a session of about one hour.

Results

The results are shown in Figures 6.1 and 6.2. Considering distance overall, the values for ↔ and ↕ do not differ from ≡ and ||| (in the paradigm experiment). □ is greater than , but not significantly.

(median test $0.1 > p > 0.05$) is possible greater than the manual trials of the paradigm experiment. This last, if true, is surprising, because the situation, apart from the surrounding experiments, is the same as that in the paradigm experiment.

The delineations were assessed for feature analysis or hypothesis testing as in the paradigm experiment. This distribution was similar to the paradigm experiment. Those rated high for hypothesis testing took the same total distance to recognition as their analogues in paradigm experiment. Those rated low on hypothesis testing (high on feature analysis), took more distance (not significantly) to recognition than their analogues in the paradigm experiment. Thus all the increase in distance in manual scans comes from those subjects mainly using feature analysis. When the results are considered by distance/letter, the simulated machine scans all give results looking like manual scans, most dense on targets, medium on non-targets, and least on blank spaces. The distances for letters are about twice those for the paradigm experiment, in all cases.

The scan densities are correspondingly higher, (median 5 for all scan conditions) than the paradigm experiment.

No subjects used the rectilinear joysticks constructively to improve a hypothesis testing scan.

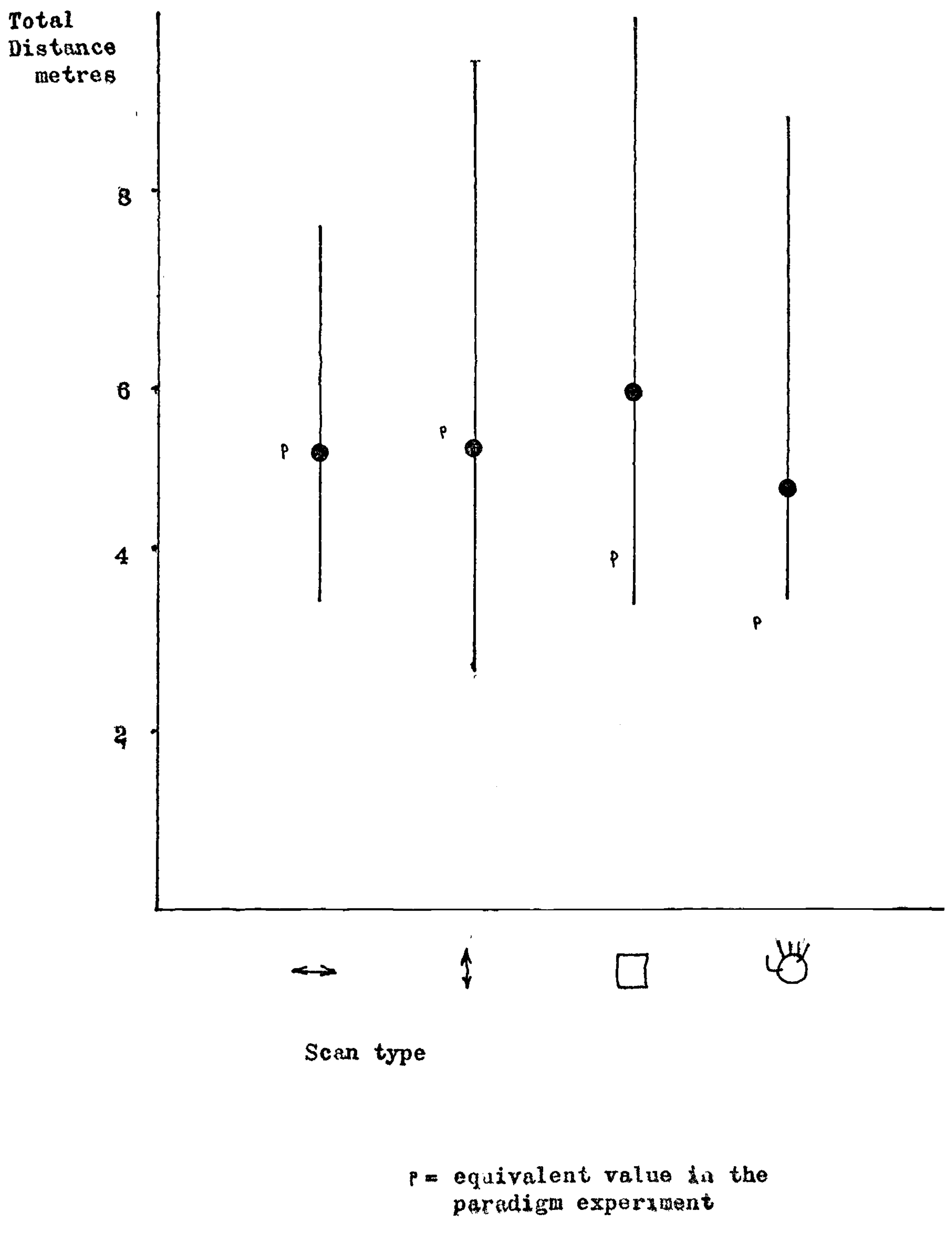


Figure 6.1

Total Distance against Scan Type

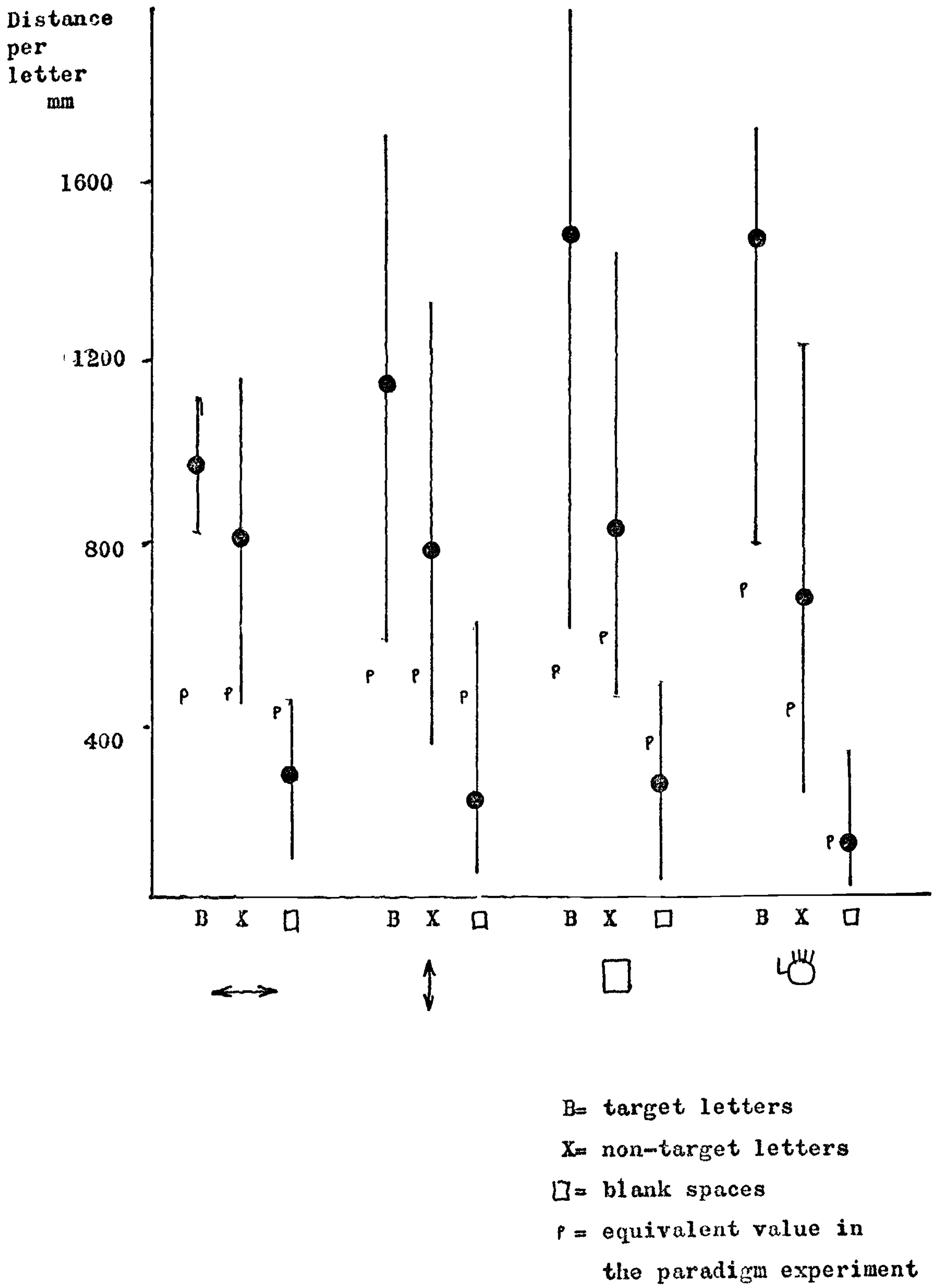


Figure 6.2

Distance per Letter against Scan Type

The fractions of target letters legible and their coverages, for all scan conditions, are similar to those for the paradigm experiment.

Discussion

Subjects gave no more indication in this experiment than in the paradigm experiment of using hypothesis testing, since no trial was rated fully hypothesis testing, and the other data (coverages, legibility, distances) are consistent with this. The subjects were not any more able to use hypothesis testing with the pair of joysticks rather than the single one, and so the absence of straight lines is not the cause of the absence of hypothesis testing in the paradigm experiment.

There is no suggestion that human control of the spacings of a machine type scan is more efficient than the uniform spacing of the machine, since the simulated machine scans took longer than the machine scans in the paradigm experiment, significantly so in distance/letter.

One argument against the utility of these experiments is that the controls were difficult to use. However, if hypothesis testing is being used, the difficult scanning would be expected to bias the subject to scan less and think more, and thus find the target letter with little scanning. If feature analysis is being used, then the difficulty of scanning precisely should make the subject scan more between attempts to recognise, since the scanning distances did increase, feature analysis again seems to be indicated.

Subjects operated the scans in the same way as manual trials, scanning letters one by one. They scanned as densely as manual trials in the paradigm experiment on all trials in this experiment. This is consistent with the scanning rule advanced before - not to scan blank areas, and to scan next to a line already revealed.

Chapter 7

EXPERIMENT WITH ALL MANUAL SCANS

Introduction

In the previous experiment subjects took more distance to recognition in manual trials among constrained manual scans than they had one on manual trials among machine scans in the paradigm experiment. This result, although not statistically significant, suggests that the tasks surrounding the manual scan trials had an effect on the style and efficiency of subjects' performance. If so, it would be useful to look at purely manual scans, to find how subjects behave without examples or interspersions, and in particular, whether the subjects in the paradigm experiment were influenced by the example of machine scans to use regular uniform scans, and whether if no alternative is suggested subjects use hypothesis testing methods.

The use of manual scan alone also allows subjects more trials in which to learn by experiment. Subjects behaviour at the end should be better adapted than their behaviour at the beginning, the rate of learning should vary with the scan type. A simple area-covering scan for feature analysis can be optimised in a few trials; a complex hypothesis testing scan is likely to need many more trials before it is perfected. Thus we expect subjects using feature analysis to learn in few, maybe 3, trials, and subjects using hypothesis testing to continue to improve over the whole session.

Procedure

The plan of the experiment was the same as that for the paradigm experiment except that all scans were manual. No instruction was given about the scan type to be used. The targets used were those in Figures 5.1 and 5.2 (targets "E" and "B"). Eleven second year psychology students, five male and six female, age range 18-21, served as subjects. None had served in previous experiments. They were paid 5/- for one session of about one hour.

Results

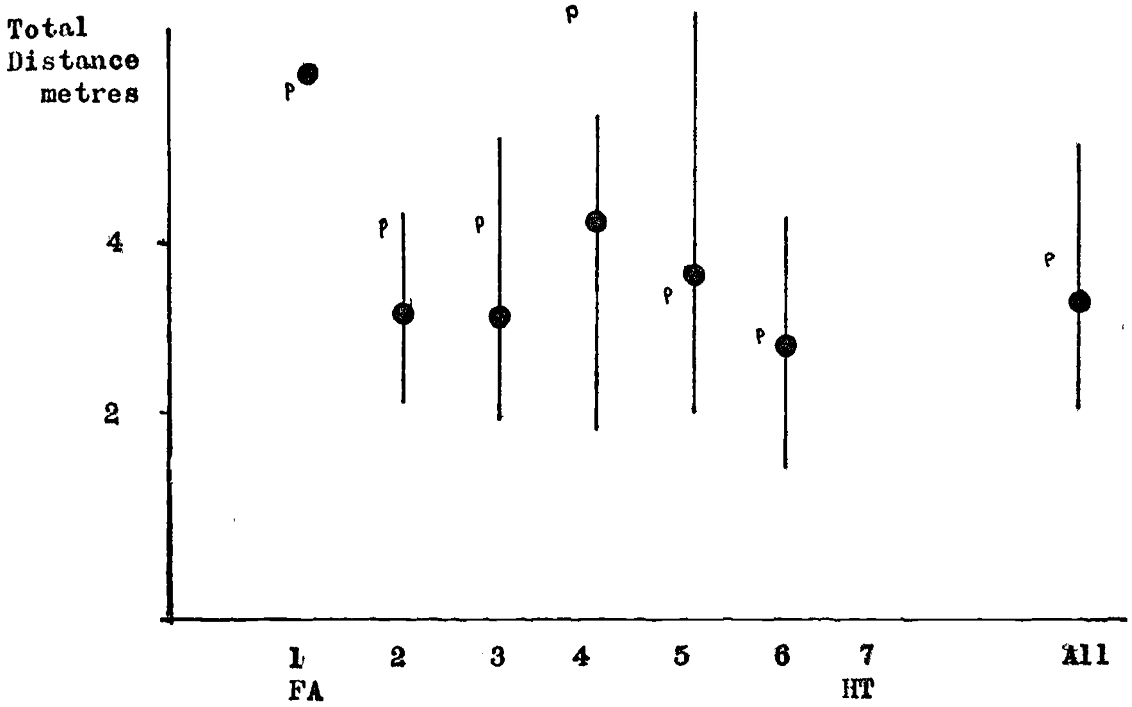
Considering total distance (Figure 7.1) there is essentially no difference in efficiency between this experiment and the manual trials of the paradigm experiment. The most apparent difference, between the two experiments, in trials assessed low hypothesis testing (high feature analysis), is not significant (median test $p > 20\%$).

Considering distance per letter, (Figure 7.2), there is again essentially no difference.

The distribution of hypothesis testing scores is essentially flat, with few assessed 1 (pure feature analysis) and none assessed 7 (pure hypothesis testing).

The letters were assessed for legibility. The results are shown in Table 7.1. Note that 93% of target letters were recognisable, whereas 55% of non-targets were recognisable. Thus again, subjects scanned letter by letter.

The performances of the first five trials by each subject, his last five trials and middle trials were compared. (Figure 7.3) There is significant improvement from the first five trials to the rest, (median test, $p < 5\%$) but no noticeable change from the middle trials to the last trials.



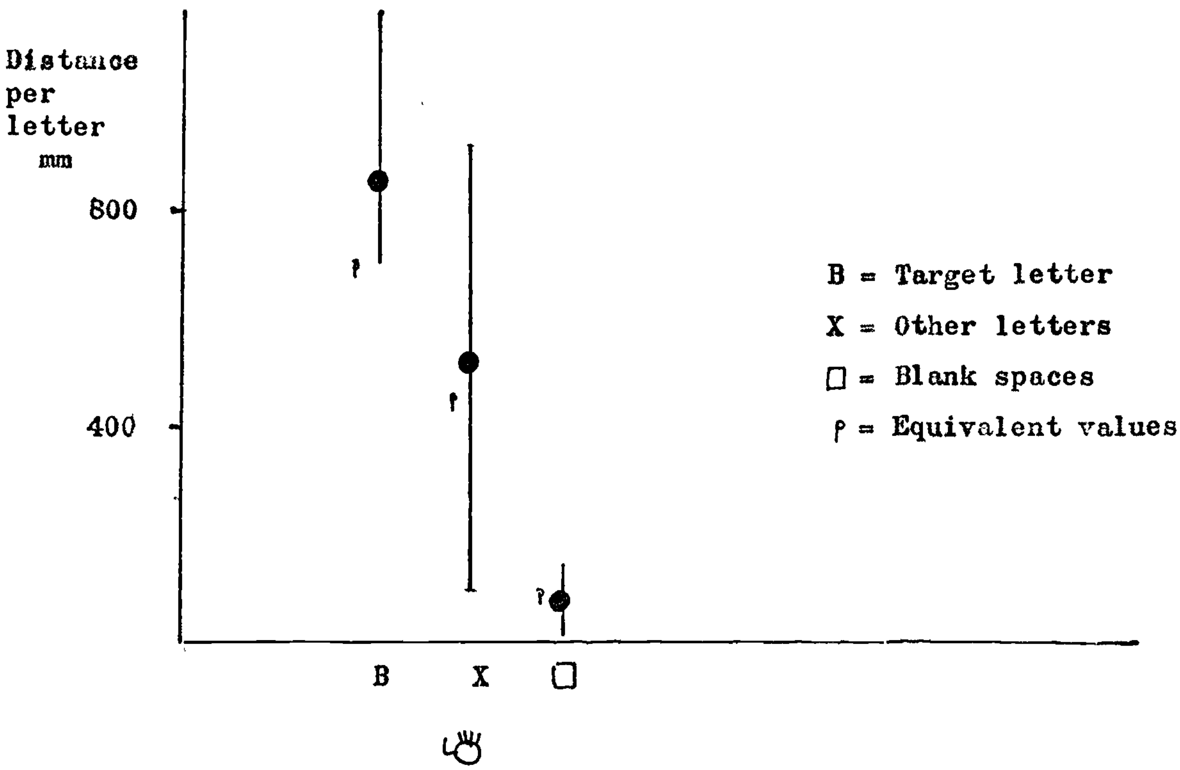


Figure 7.1

Total Distance to Recognition

Figure 7.2

Distance per letter

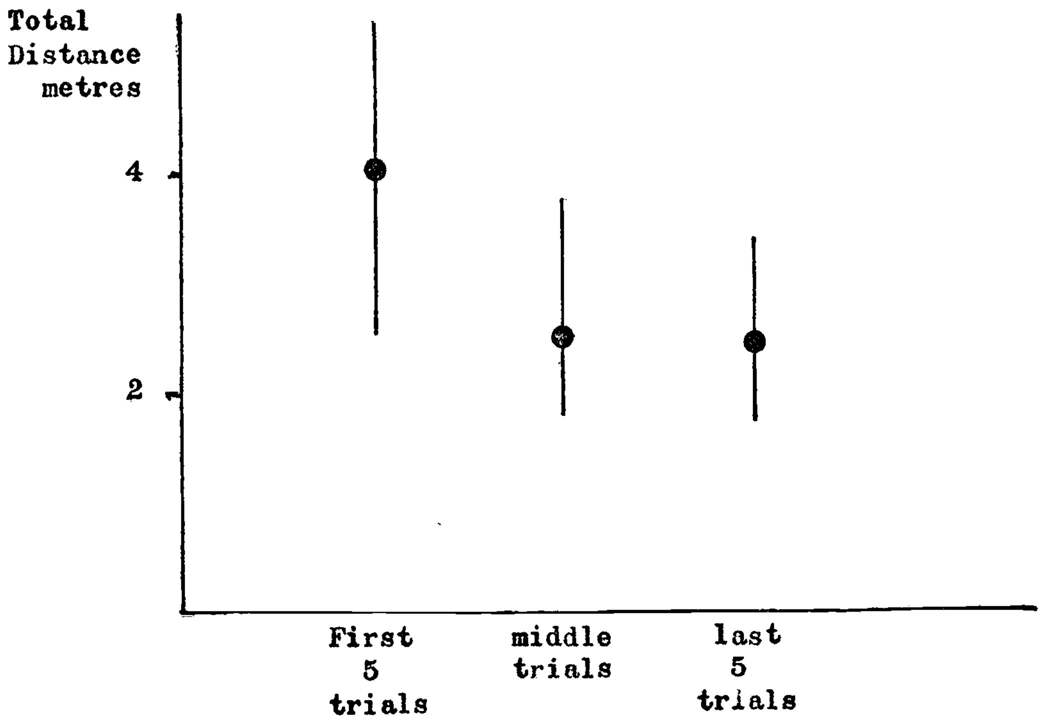


Figure 7.3

Learning effect

Assessed FA-HT	Target letters			Non-target letters		
	Readable	Deducible	Illegible	Readable	Deducible	Illegible
1 (FA)	100	0	0	57	7	36
2	87	8	5	42	16	42
3	88	7	5	49	19	32
4	74	26	0	37	18	45
5	79	14	7	39	14	47
6	57	28	20	16	30	56
7 (HT)	-	-	-	-	-	-

Table 7.1

Assessed Feature Analysis - Hypothesis Testing and Legibility

Discussion

The similarity of results of this experiment and the manual trials of the paradigm experiment implies that the latter were essentially uncontaminated. Subjects are not induced to feature analyse by the machine scans interleaving the trials, but that they were confused by the multiple joysticks of the simulated machine scans, and that this confusion carried over to their use of a single joystick in the manual trials of the previous experiment. The subjects did not improve their scanning by experience, after the first five. This is what might be expected of feature analysis, but not of hypothesis testing, where improvements should be available for a long time.

Chapter 8

EXPERIMENT WITH VARIABLE TARGET LETTERS

Introduction

In the introduction it was pointed out that hypothesis testing ought to be particularly efficient when the target letter is held constant, because an efficient scanning strategy to match this one letter can be evolved. Thus hypothesis testing would be more efficient with a constant target than when used to find different target letters. Feature analysis is little affected by the change of target letter, because the scan is continued for any letter until it is identifiable, which will be at a similar scan density for all letters.

If hypothesis testing is being used, then there should be an effect when different target letters are used in different trials: if the subject is not using a strategy designed to match only one letter, it matters little what letter it is.

Procedure

The transparencies used were those in Figures 5.1 and 5.2, used in batches of five with the same number of letters per frame. The target letter was chosen from among those on the frame by throwing a dice. The subject was informed of the target letter for each frame before the presentation of that frame. The task and equipment was otherwise the same as the third experiment, using both manual and machine scans.

Nine second year psychology students served as subjects, five male and four female, age range 19-22. None had served in previous experiments. They were paid 5/- for one session of about an hour.

Results

The results of this experiment are very similar to the paradigm experiment, both considering distance overall (Figure 8.1) and distance per letter (Figure 8.2).

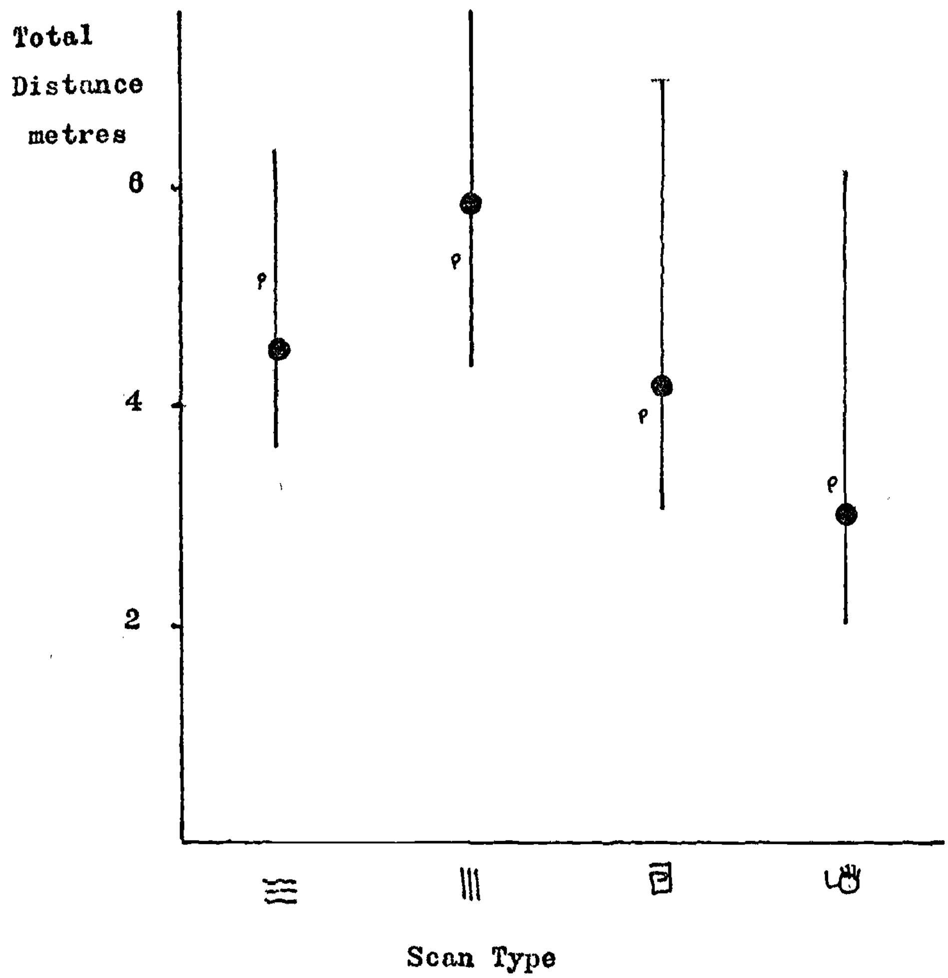
The ≡ scan is more efficient than the |||, whereas in the paradigm experiment ||| was slightly more efficient, although the difference is not significant. This is reasonably explained, because both "E" and "B", the previous target letters, had characteristic signatures to a vertical scan (3 dashes) and were easily recognised by this. Other letters do not have such simple features laid out vertically, and so will not be so simply recognised by a vertical scan. The difference in total distance to recognition between high and low to hypothesis testing manual scores has now vanished. The distribution of hypothesis testing scores is no longer flat - more trials were assessed at the feature analysis end, and fewer at the hypothesis testing end.

The letter legibility is as for the paradigm experiment, but with more non-target letters recognisable.

Discussion

The distances to recognition are the same as for the paradigm experiment. This implies feature analysis is the normal method of recognition, as under hypothesis testing a change of target would lead to an increase in distance to recognition because the strategy would have to be adapted for each new target.

The changes which did occur, the shift of assessments of hypothesis testing towards feature analysis and the decline of



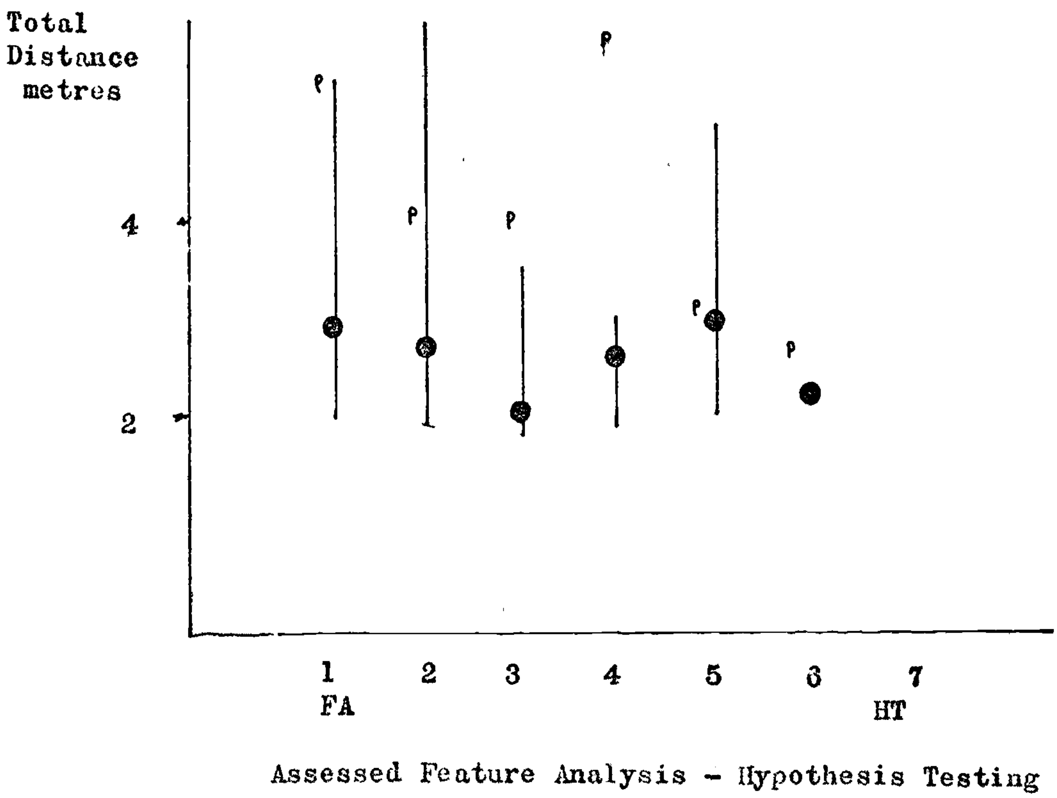
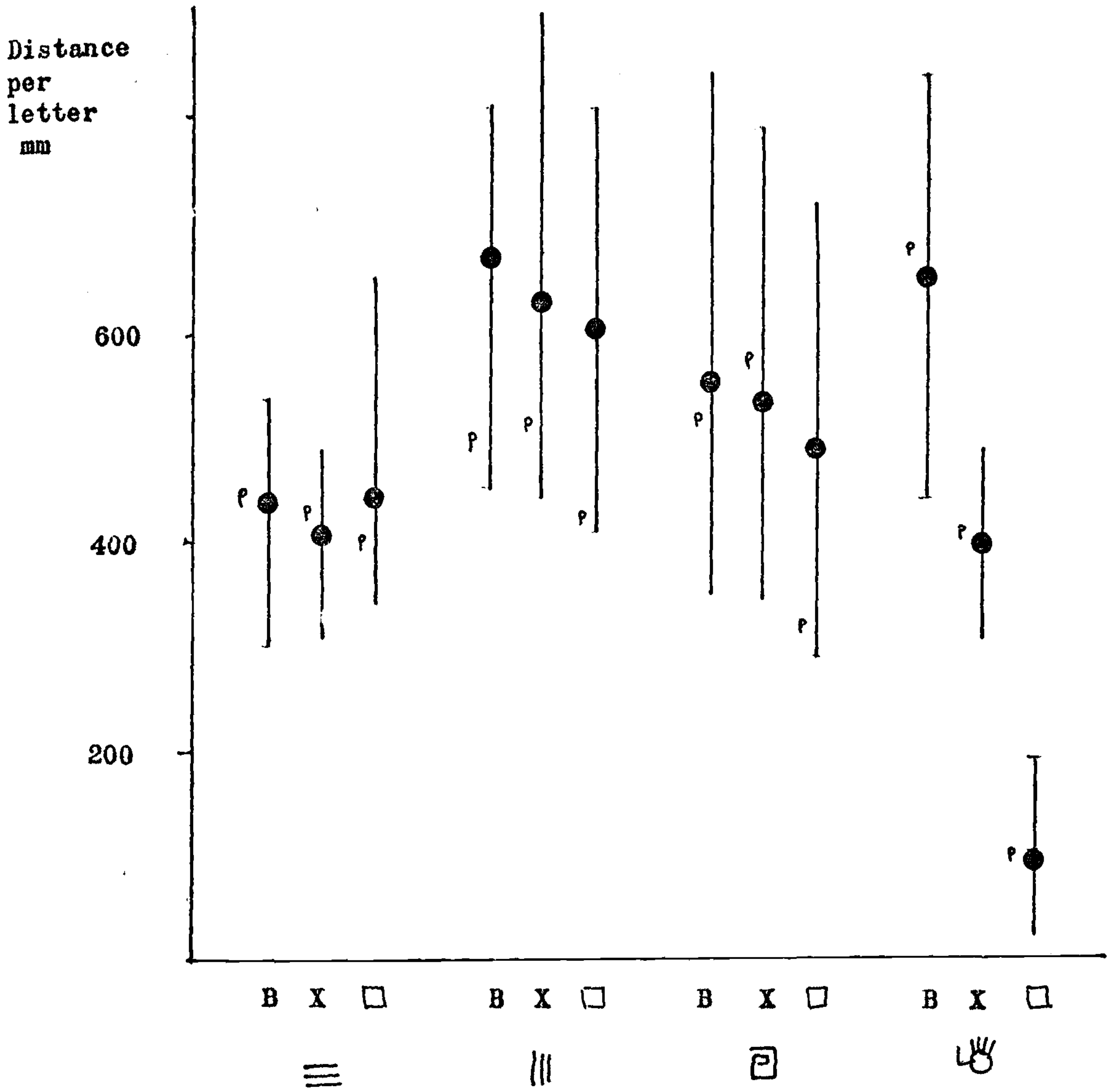


Figure 8.1

Total Distance to Recognition



B= Target Letters

X = Other Letters

□ = Blank Spaces

P = Paradigm Equivalent

Figure 8.2

Distance per Letter against Scan Type

efficiency of trials assessed high hypothesis testing, are as would be expected if the utility of hypothesis testing had been much reduced for the few subjects who used it.

Chapter 9

A: EXPERIMENT WITH IDENTIFICATION

Introduction

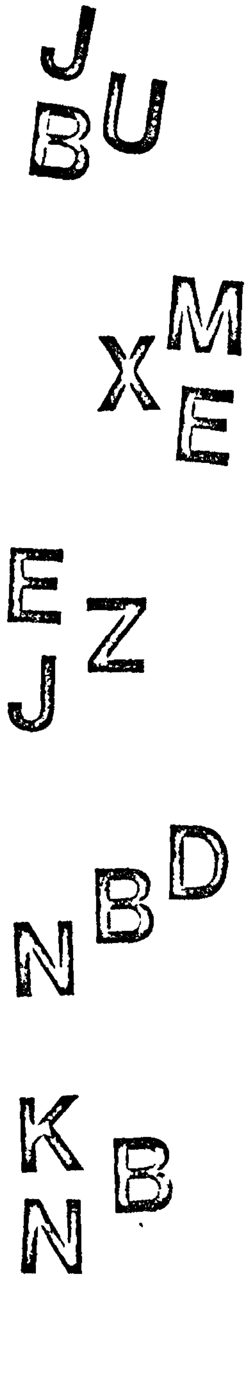
The paradigm experiment and its variants show that subjects rarely use hypothesis testing methods of pattern recognition.

They were apparently scanning letters one by one, and identifying each by feature analysis. They did not appear to stop at a point where the letter was scanned enough to be clear that it was not the target, without knowing more precisely what it is; that is, they were answering "What letter is it?", not "Is it B?". In this case they should need no extra scanning if their task is changed to identifying the letter explicitly. Subjects using hypothesis testing should increase their scanning when asked to identify the letter, because more hypotheses need to be produced and tested in identification, whereas matching only requires the original hypothesis which was given. Thus if subjects use feature analysis, there should be no change if they identify rather than match letters, whereas if they use hypothesis testing then there should be an increase in scanning for identification compared with that for matching.

Procedure

The transparencies used were those with three-letters-per-frame shown in Figure 9.1 and a new set consisting of a whole alphabet of single letters from the same size and typestyle, set in the middle of the frame. The subject's task was to recognise every item in the frame.

Nine second year psychology students, two male and seven female, age range 19-22, served as subjects. Only one had served



and an alphabet of similar letters set in the centre of the frame.

Figure 9.1

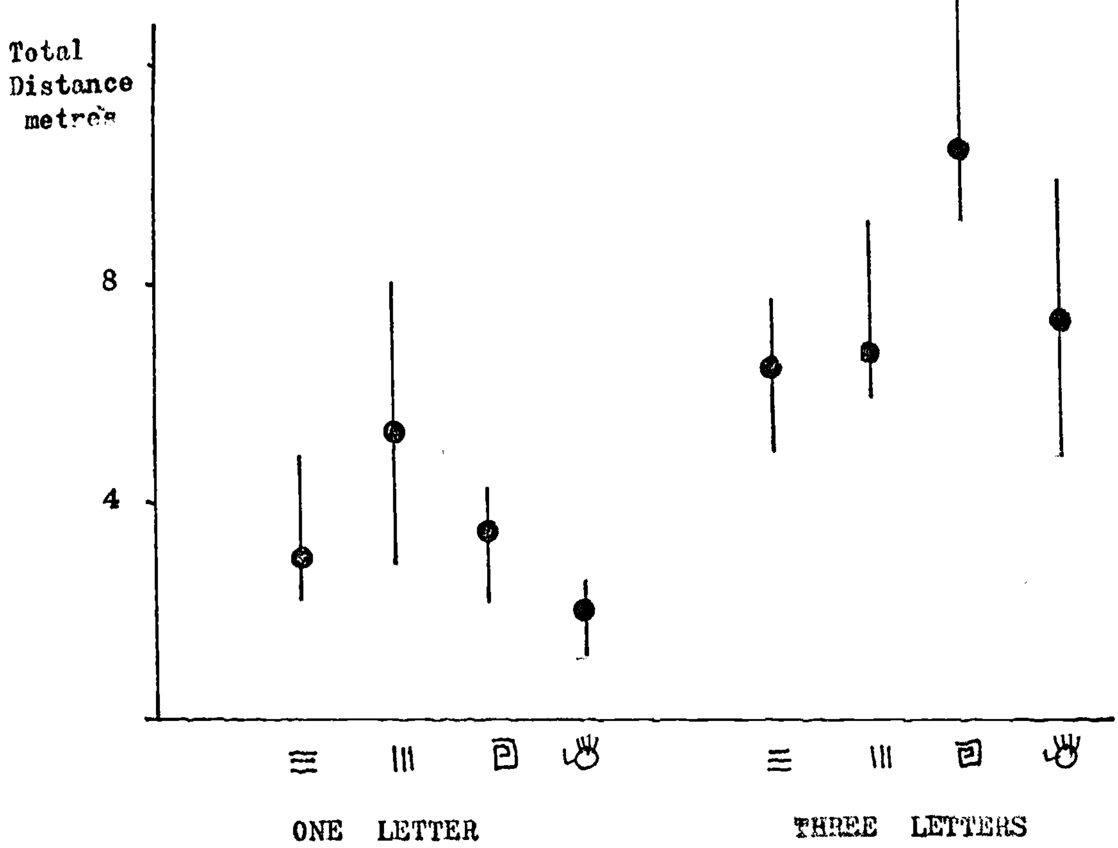
Letter targets for Recognition

in a previous experiment. They were paid 5/- for one session of about one hour.

Transparencies were presented in two batches of five; a batch of three-letters per frame targets and a batch of five randomly selected single letters. The order of these batches was randomised. These two batches were presented using the same scans as before (≡ , ⦀ , ⧉ , ⋓ , ⋓ ,) randomly ordered.

Results

The total distances to recognition are shown in Figure 9.2. The machine scan results are not significantly ($p > 5\%$) different from the equivalent values for the paradigm experiment. The manual values for one letter per frame are less than those for the paradigm experiment, which is an unexpected direction; the difference is not significant. The manual values for three letters per frame are greater than those for the paradigm experiment ($p < 1\%$). (Subjects were asked to recognise all three letters in this experiment.) It is also quite compatible with being 3 times the value for one letter per frame in this experiment (median test on distances for one letter per frame, and distances for three letters per frame divided by 3, $p > 75\%$). In the paradigm experiment, they only needed to recognise 1.6 letters on average before reporting, which explains this increase in distance. 88% of the letters were immediately recognisable and 12% were deducible. None were illegible. This suggests that the subject and the experimenter as assessor of legibility are working on the same data, and have the same standards, because both find 100% of the letters legible. This is a higher proportion of recognisable letters than in target letters in the paradigm



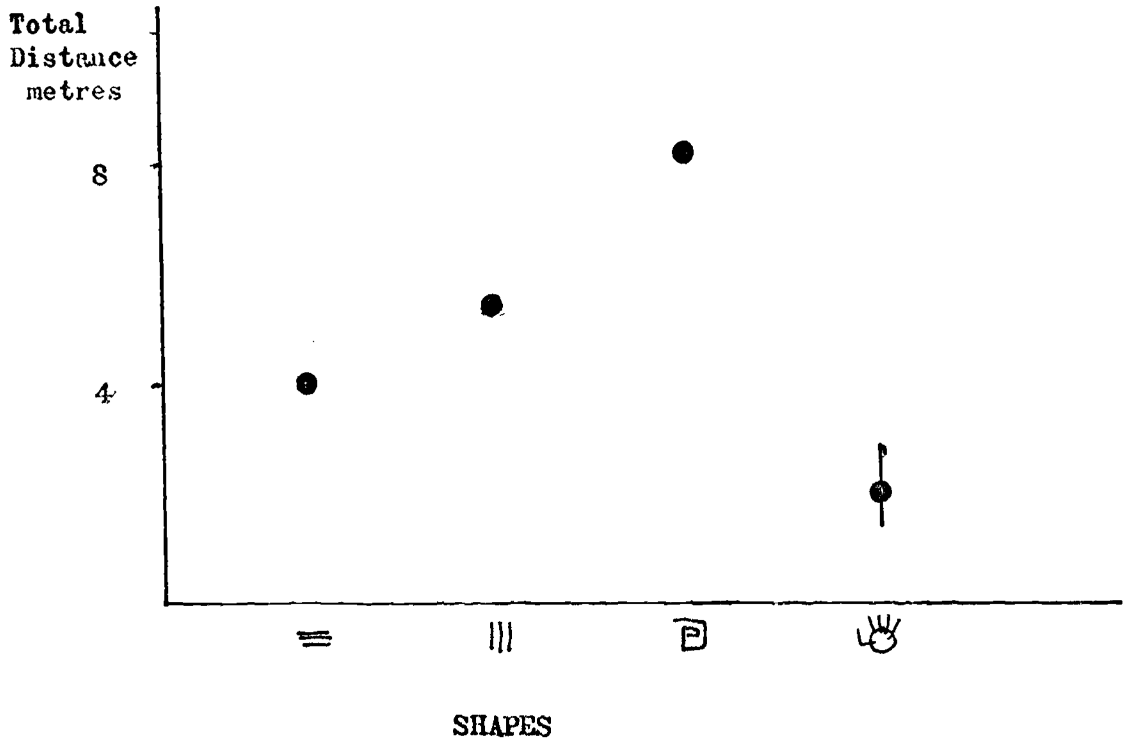


Figure 9.2

Distance to Recognition for Letters

Figure 9.4

Distance to Recognition for Shapes

experiment, where, with a frame of two letters, if one was recognised as not the target letter, this implied the other was the target letter without any scanning. This produced an illegible target letter, and lowers the proportion of recognisable target letters.

The distribution of feature analysis-hypothesis testing did not differ significantly from that in the paradigm experiment.

The distances to recognition for machine scans are not significantly (median test $p > 10\%$) different from their corresponding values in the paradigm experiment.

Discussion

The similarity of distance to identification rather than matching shows that the different tasks in this and the paradigm experiment caused no change in subjects behaviour, for the increase in distance for three letters is predictable from subjects having to scan all three letters, as explained in the results. It was not that identifying was more difficult, just that there was more to do. The similarity with the paradigm experiment for both one and three letter frames with machine scans corroborate the results of the paradigm experiment.

All these results confirm the paradigm experiment conclusion that subjects were always identifying letters by feature analysis.

B: EXPERIMENTS WITH SHAPES

Introduction

More information about subjects recognition methods may be gained by using shapes instead of letters.

Feature analysis demands dense enough scanning of the picture to clarify its form. The amount of scanning that delineates a letter clearly would delineate a shape of the same height and width equally clearly.

Hypothesis testing demands scanning to test hypotheses. With the limited set of letters as hypotheses, this means rather little scanning. With an unrestricted set, that of all shapes, the number of hypotheses tested before success, and the amount of scanning, could be large.

Matching to a set of shapes is difficult, because it introduces the bias of an external list, in that it "costs" the subject effort to consult the list and thus biases him towards feature analysis.

An alternative, which does not introduce this bias, is to have the subject identify simple nameable shapes without a list (eg, cross, diamond).

Thus, using a set of shapes of about the same size as the letters, as targets to be identified if feature analysis is used, then the scan length should be the same as letters, and if hypothesis testing, then very much more. Identification of shapes, and the identification of letters, are tests of feature analysis or hypothesis testing which are independent of the experimenter's judgement of the delineations.

<u>Procedure</u>

This experiment was run concurrently with 9A, the shape trials after the letter trials.

5 shapes were presented with the usual scans (≡ , ⦀ , , ,) in random order, then 5 shapes with purely manual scans.

The subject was told that the shapes could be described in one word, but no other restriction was put on their range. The

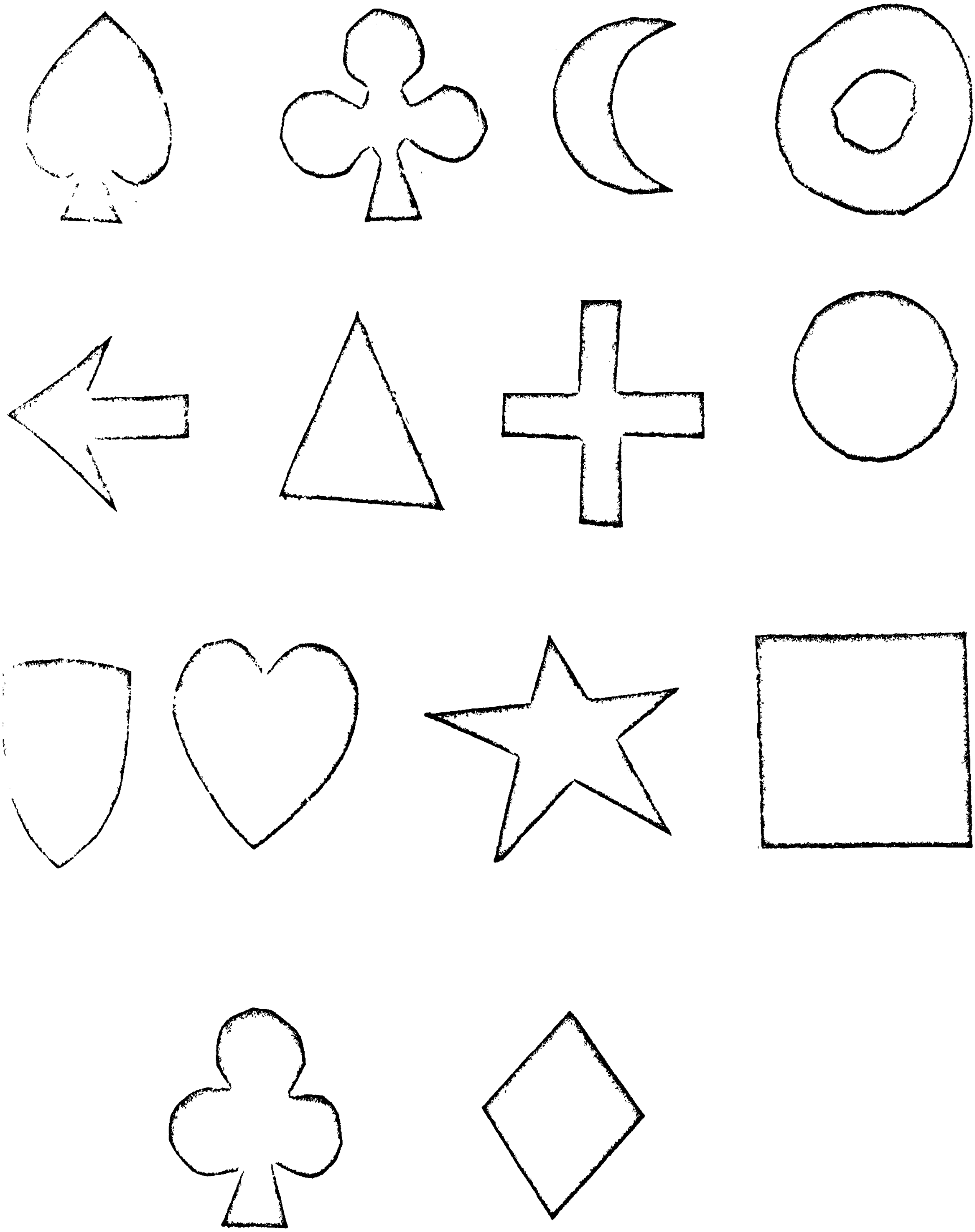


Figure 9.3

Shape Targets

shapes are shown in Figure 9.3. Subjects were asked to identify the shapes.

Results

The results for shapes, shown in Figure 9.4, are very close to those for single letters in Figure 9.1, and to those for letters matched in the paradigm experiment. The exception was the scan, where recognition of shapes took more scanning (twice as much) than for letters (median test, $5\% < p < 10\%$).

The assessed feature analysis or hypothesis testing of shapes was biassed to the feature analysis end, compared with that for letters, ($p < 10\%$, K-S) and compared with a flat distribution ($p < 1\%$ K-S). The scan densities were similar to those for letters in the letter identification trials.

This implies that subjects could recognise the shapes easily as letters.

Discussion

Subjects identified the shapes as easily as the letters.

Under feature analysis, the delineation will be recognisable when the scanning is dense enough, independent of the expectedness of the pattern. Under hypothesis testing an unexpected pattern should take more scanning than an expected one because the subject is likely to test and reject many more guesses. This experiment shows that varying the expectedness of the target does not change the amount of scanning needed, which is consistent with recognition by feature analysis.

Chapter 10

COMPUTER SIMULATION

Introduction

This study is based on the possibility of hypothesis testing. The evidence from the experiments is that subjects do not generally use this. If this is because it is impossible, then the study is pointless. A few trials were solved by hypothesis testing, but stronger evidence for the validity and usefulness can be produced if hypothesis testing can be used to recognise letters in a computer based simulation of the task.

Simulating one recognition model alone is risky, in that the simulation may be more or less efficient than the original, because of the programming and the special properties of the machine. Comparing models allows us to separate the effects of the computer and the intrinsic properties of the model.

For useful comparison the alternative model should be feature analysis, because it has been postulated as the theoretical antithesis to hypothesis testing and because the evidence from the experiments is that subjects used some form of feature analysis model. Simulation of potential feature analysis models therefore explores possible models underlying subjects' behaviour. The simplest form of this is stencil matching; this seems very crude. It will be shown to be inadequate for the task in this study because it is too sensitive to ignorance of the position and size of the letter. Therefore a more complex feature analysis model of recognition, based on horizontal slices of the delineation, is postulated as an estimate of what the subjects do.

Three models of letter recognition were simulated, using as input scanning analogues to that in the experiments.

1. Two variants of a hypothesis testing model, aimed at deciding whether the letter was an "E", a "B", or something else, with as little scanning as possible.
2. A stencil matching model. This is an unsophisticated program of which a visualisable analogue is putting a letter of the same size and position down over the delineation, and measuring how much of the delineation is outside this. The correct letter will have the smallest excess.
3. A more sophisticated model, based on horizontal "slices", working on the relations of features.

The hypothesis testing model drove the scanning itself, the other two models scanned by some simple rule, and then attempted to recognise the resulting delineation.

It is not intended that these programs be very accurate letter recognisers in their own right. The input is given in a form that makes this unlikely. The delineation does not differentiate between blank areas and inscanned areas, and the sampling points must be closely spaced which means there may be many of them, implying a corresponding increase in processing time.

It is intended that the programs be models of potential human processes, and that where they parallel actual human processes they give similar results.

Procedure

Two systems co-existed, scanning and recognition. The scanning system held simulations of the letters and scans, and output a delineation of the letter. This is analogous to the experimenter and his system. This part was common to all the simulations.

The recognition programs worked only from the delineation. They were not passed knowledge of the size and position of the letters, nor, of course, which letter was being scanned.

The programs are written on POP-2. (Burstall et al, 1968)

The delineation was represented on a 100 * 100 field, although data was not quantized to whole units.

<u>The Data System</u>

<u>Conventions</u>

A capitalized word refers to the computer program:

a word in double brackets ((FILE)) refers to a section of program.

a word in single brackets (WORD), or no brackets WORD, refers to a word in the program.

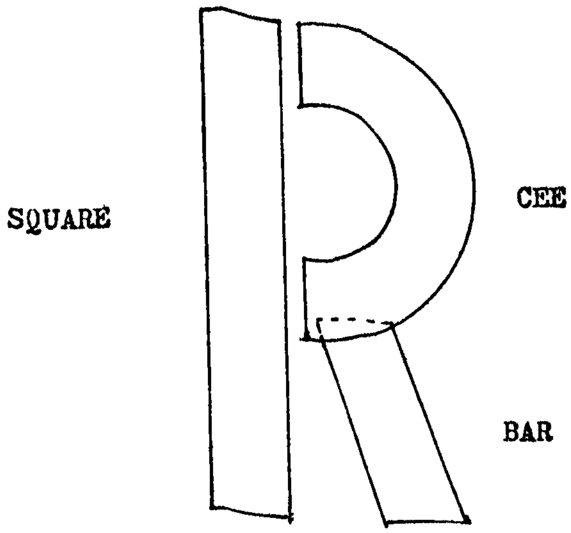
((LETTERS))

Letters were represented by a function of a point which was true if that point was black. Letters were defined as a combination of functions representing rectangles (SQUARE), quadrilaterals (BAR), and "C" shapes (CEE). This is shown in Figure 10.1. The function could represent any letter of any size anywhere on (or off) the field.

Representative letters are shown in Figure 10.2.

((MOVES))

The letters were scanned by a basic function MOVE, which went from the present position to a specified position,



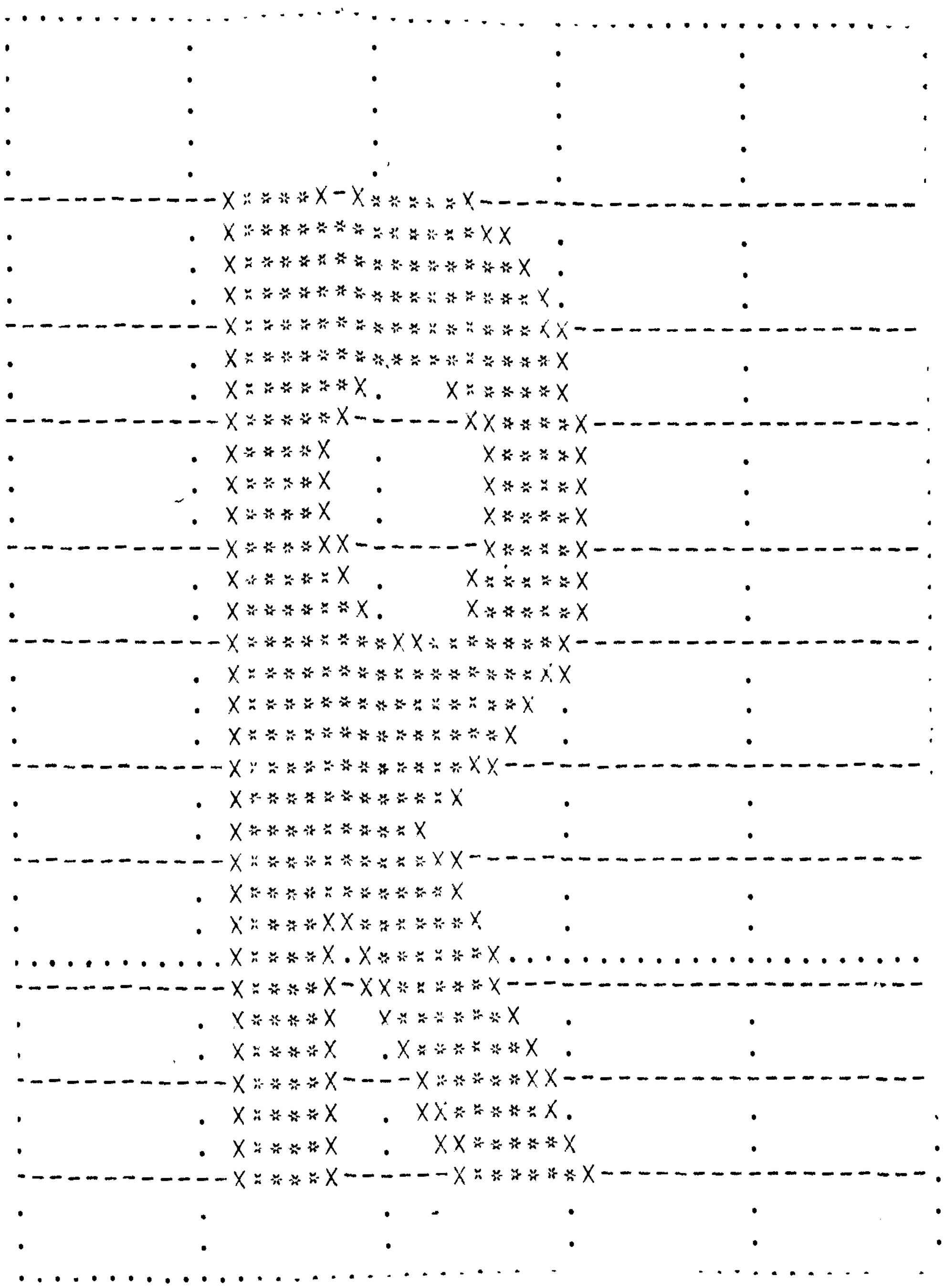


Figure 10.1

Assembly of Simulated Letters

Figure 10.2

interrogating the letter function at unit intervals, putting points found black into the delineation (POINTLIST), and keeping statistics.

Four scan functions arranged this into scans simulating those used in the paradigm experiment. They were arranged so that when called they scanned for given distance, and stopped. They were XSCAN (≡), YSCAN (⦀), HATCH ( ) and SCRIBBLE, which moved from the present position to a random point outside the extremities of the letter. This was intended to represent a manual scan.

The stencil matching and "slices" programs called a scan function, attempted recognition, and, if unsuccessful, scanned some more and repeated. The hypothesis testing program controlled the scan itself.

((DISPLAY))

The delineation and the workings of the recognition program could be displayed.

((GENERAL))

ATTEMPT allowed a whole alphabet to be tested with one instruction.

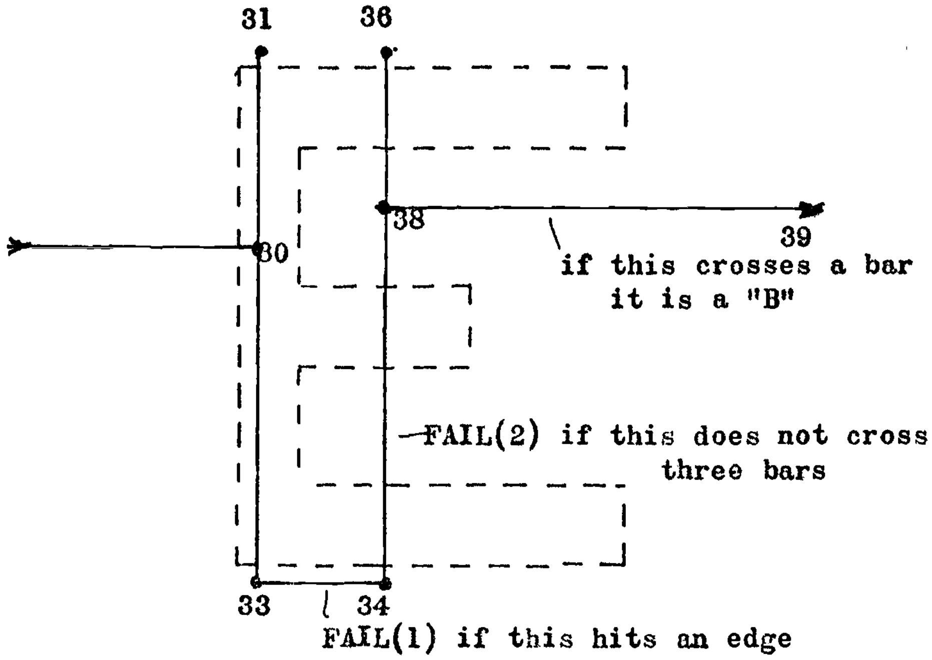
Hypothesis Testing Model

The basis of the hypothesis testing method is that a guess is made about what the letter is, and a scan made to confirm or deny this. In the paradigm experiment the task was to find an "E" or "B" - the guess is given. In the simulation this implies that the letter is hypothesised to be an "E" or "B". The critical features of both are a straight leading edge and three horizontal bars. The differentiating features are the loops in the "B". Scans must be made to confirm these features, and to reveal any discrepancies from them. The model matches two letters, rather than identifying them out of 26. This simulation is an example of a hypothesis testing strategy.

To start, the field is scanned coarsely until something is found. This is hypothesised to be an "E" or a "B", and tests which could falsify this hypothesis, are applied by scanning round parts of the figure in the expectation of finding particular things. If these are not found then the letter is not "E" or "B". No attempt is made to find what else it is. Other hypotheses (target letters) would necessitate a different set of scans and expectations.

The actual scans used are best seen from diagrams, Figures 10.3, 10.4. They test all the features that should be there, and terminate if something contradictory is found. The features looked for are:

1. Straight left edge
2. Straight bottom edge
3. Three horizontal bars
4. Straight top edge



NB Numbers are line numbers in ((GREGORY))

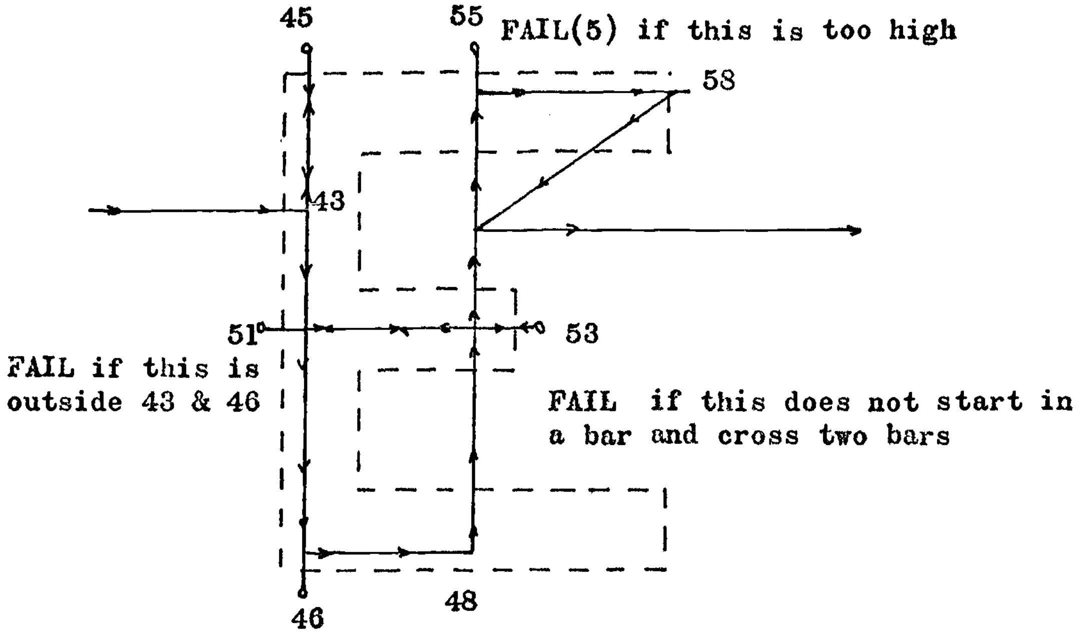
29 - 33 Show the height of the letter. If the edge is not straight and vertical, this will give a false value and cause later tests to fail.

33 - 34 Move horizontally 1/3 height. If it hits an edge, the leading edge was no straight or vertical.

34 - 36 Move vertically across three bars.

38 - 39 Moves down 2/10 of letter height, and then right, to test for the closure of a "E".

Figure 10.3
Operation of EGREG1



NB Numbers are line numbers in ((GREGORY)).

43 - 46	Find the height of the letter. A wrong value causes later tests to fail.
46 - 48	Move 1/5 height horizontally
48 - 49	Move vertically to the end of the second bar.
49 - 50	Step back into the bar.
50 - 51	Move left to an edge.
51 - 52	Move back into the letter. If 52 is the leftmost point discovered, the edge is not straight.
52 -54	Move right to edge, then step back in.
54 -55	Move back to the original vertical track and up. This should cross two bars.
55 - 57	Step back into the bar. If this is higher than 45, then the top edge is not straight.
58	Move to the end of the letter.
59	Move back to the vertical track, 1/5 of letter height down. Move out to test for loop of "B".

Figure 10.4

Operation of EGREG2

Thus, if features 1-4 are found, it is an "E" or "B". These can be differentiated by searching for the enclosing bar on the right hand edge of "B".

This scan has the properties required of hypothesis testing - that the scan is controlled on the basis of the information already revealed. The scans are terminated exactly at edges, moved fractions of the letter height, and abandoned when the letter is shown not to be a target.

Human subjects may very well not stop where the computer fails, but change their hypothesis and continue. This is redundant within the paradigm experiment, but useful in the identification experiment.

Results

EGREG1 did not reliably differentiate "E" from "Z". The results given are for EGREG2. The system was set to work on four alphabets, differing in size and position, and made two errors.

The distances to recognition were small, one tenth those for the "slices" recognition system, with a median distance to recognition of 3 letter heights, equivalent to 100 mm in distance per letter for the human experiments - about the same as human manual scan performance on blank areas, which were apparently scanned only enough to reject them. Six (out of 128) targets manually scanned in the paradigm experiment took less distance per letter than this to recognise, and some of these may have been recognised by exclusion, by identifying all the other letters in the frame.

Table 10.1

Results of hypothesis testing model, with varying size and position of letter.

EGREG2

XLO *	YLO *	SIZE	Errors in one alphabet
20	20	61	-
5	5	60	B reported as E
10	10	88	B missed
20	20	77	-

Part alphabets of six other sizes (64 letters in all) were also run, and recognised correctly.

* Coordinates of position.

Feature Analysis Models

The task in this simulation is, as in the paradigm experiment, to recognise a delineation of a letter.

If feature analysis is being used, scanning is not indicative of the recognition process, and only serves to introduce a variable amount of information to the recogniser.

The feature analysis models operate after the image has been scanned; they convert the delineation into an internal code which is matched against a stored representation. The abstractness of this representation ranges here from low in the stencil matching model, to medium in the slices model. As the abstractness of the representation increases its generality increases, and the amount of work needed to match it to the internal representation decreases, while the amount of work needed to generate the representation from the delineation increases.

The stencil matching model

(STENCIL)

A supervisory function (STENTEST), went through the alphabet, setting each letter up (LETTER), scanning it a small amount (SCAN), with any of the four simulated scans, and applying the stencil matching function (STENCIL) to the delineation.

The stencil matching function first found the size and position of the letter (LIMITS). The size is the maximum vertical separation of points in the delineation (HEIGHT).

It then went through the alphabet, setting up letter functions of that size and position (LETTER), and finding the number of points in the delineation (POINTLIST), which were not in that letter (ERRORS).

If the letter was not recognised then it was scanned some more, and (STENCIL) reapplied to the new delineation. If the letter had been recognised, or if the letter had not been recognised after a lot of scanning then the attempt was stopped, and the next letter started.

Results

The results of this process are bad. The machine recognised only three letters out of eleven presented (at least two were presented with each machine scan), taking about half an hour of computer time per trial. This is not an adequate simulation of subjects' behaviour.

The difficulty was that the estimate of size and position was usually wrong. The trials were successful when the estimate of size was within 1% of its correct value, which only occurred with random scanning (SCRIBBLE).

Thus this stencil matching model is very sensitive to wrong information about size and position. It is not a good method to use for recognition from scanning as scanning rarely gives good information about the extremities. It also cannot discriminate between letters differing only by an addition ("E" and "F", "Q" and "O"); because it counts points outside the letter.

The "slices" Model

Stencil matching was inadequate; other recognition mechanisms must be sought.

To be a simulation of human recognition the process should have the following properties; it must recognise delineations of letters by feature analysis. It should produce the same results as human recognition of machine scans. (The large amount of scanning in manual trials is probably due to the deficiencies of humans as scan generators, not as recognisers.) It should need most, but not all of the letter for recognition, which excludes some variants of non-relational feature analysis. It should be biassed to the rectilinear - diagonal scans took more scanning. This is not so of simple stencil matching, for instance - it should only need moderate scan densities, rather than the extreme densities that would be needed for micro-feature analysis.

More generally, it should be a normal letter recognition mechanism which extends to this specialised scanning task. If the system can also be implemented as an efficient machine recogniser, so much the better.

The "slices" model that is simulated is one that meets these criteria, but is not unique in this.

Background

The mechanism simulated is of a vertical line of detectors scanning along the line of print. When all the detectors signal white they are at a space between letters, (or between words, which is not the problem under consideration). Then, from the order of reporting black the shape of the leading edge of the letter can be found; if "A" shaped, the bottom detector will signal black first, with detectors above it signalling later. Similarly, the shape of the trailing edge can be found. In between, the letter can be specified by whether the detectors meet long elements, horizontal bars, or a number of short elements, vertical bars (normally one line width, although some lines will be crossed obliquely).

While the concept of the simulation is a line of photodiodes scanning a piece of paper being moved underneath, it is not essential that the mechanism be sequential. The same system can operate with spatially distributed detectors acting in parallel, and 'to the left of' acting instead of 'before'. Thus, where in the sequential case all the detectors indicating white at one time implied a gap between letters, in the parallel case one row of detectors across the print line, all indicating white implies a letter gap, and a row of white-black boundary detectors indicating positively, each above and to the left of the next, indicates a "V" leading edge. Thus the discussion will be in the form of a sequential model, but, with the appropriate changes and a huge amount more equipment, it is equally appropriate to a parallel acting spatially distributed model.

The data required for the final matching are relative positions of parts of the letter, rather than absolute positions within the letter; for example, an "E" has three lines above each other, not lines at specified heights. This means that the process is relatively insensitive to changes in typestyle, and can easily be adapted to different heights, widths and positions of letters.

<u>Procedure</u>

((SLICES), (DICT), (GENERAL))

In this, as in the stencil matching simulation, the letter was simulated and scanned (ATTEMPT). The recognition program (INSPECT) was then applied to the delineation. If the letter was not recognised it was scanned some more and recognition was re-attempted, unless an upper limit of scanning was exceeded, when the process was stopped.

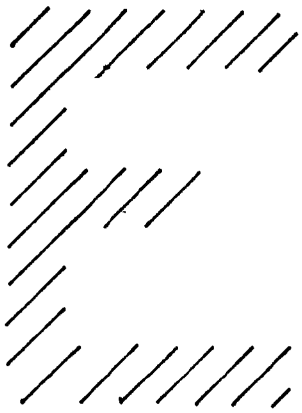
The workings of the recognition program are shown in Figure 10.5. The top and bottom parts of the leading and trailing edges were analysed separately. The edge was "I" if all the leading points lay within a small distance of each other, "A" if they all lay above and to the right of each other, "V" correspondingly left and "S" otherwise. 'I', 'A', 'V' and 'S' reasonably enough gave leading and trailing edges of the type denominated by themselves. These edge symbols and the list of line lengths shown in the figure made up the analysed description.

The entries in the dictionary, (DICT) in the same form as the analysed description, were set by the programmer. The description and the dictionary entry had to match exactly for recognition - no alternative entry scheme was used. The maximum allowed scan distance, the scan distance between recognition attempts, the height and placement of the letters, the number of horizontal bands and the spacings of the scans were all set to reasonable values. Two parameters required tuning - the distance between points which corresponded to a gap, and the length of a line that was "long".

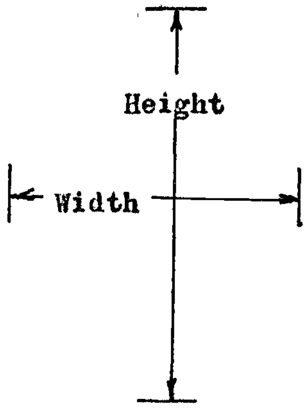
The system was set to recognise every letter of six alphabets, with various sizes and scans. These are shown with the results in Table 10.1.

Results

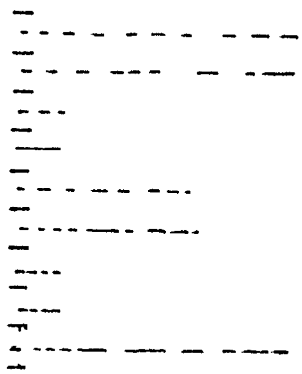
The results are shown in Figure 10.6 and Table 10.1. The distances to recognition are intermediate between the stencil



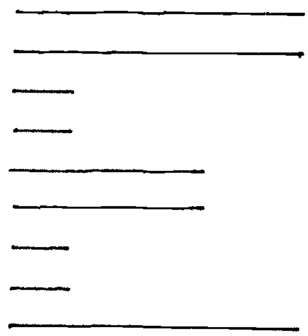
1) Original Delineation



2) Find size



3) Dissect into slices



4) Join the slices

L
L
l
l
L
L
l
l
L

5) Translate slices

6) (L l L l L)

7) Leading edge "I" "I"
Trailing Edge "S" "S"

8) I I S S (L l L l L)

9) "E"

The process of recognition is ;-

1) Scan to give a list of black points (POINTLIST)
2) Find the extreme top, bottom right and left points (LIMITS)
3) Split the height of the letter into 9 horizontal bands and combine points lying within a band to points lying al along a line (FILLSLICES)
4) Combine neighbouring points along a line to give line segments. (ENCODE)
5) Count the line segments in each line. If one is long put "L", not the number. (ENCODE)
6) Form a list of these values, and delete items until no adjacent values are equal (LINEFIND)
7) Find the shapes of the leading and trailing edges (EDGESORT)
8) Combine all this information, and look it up among entries of the same form in the dictionary (INSPECT).
9) If a letter is recognised, stop; otherwise repeat from 1).

Figure 10.5
Operation of ((SLICES))

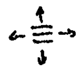
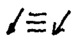
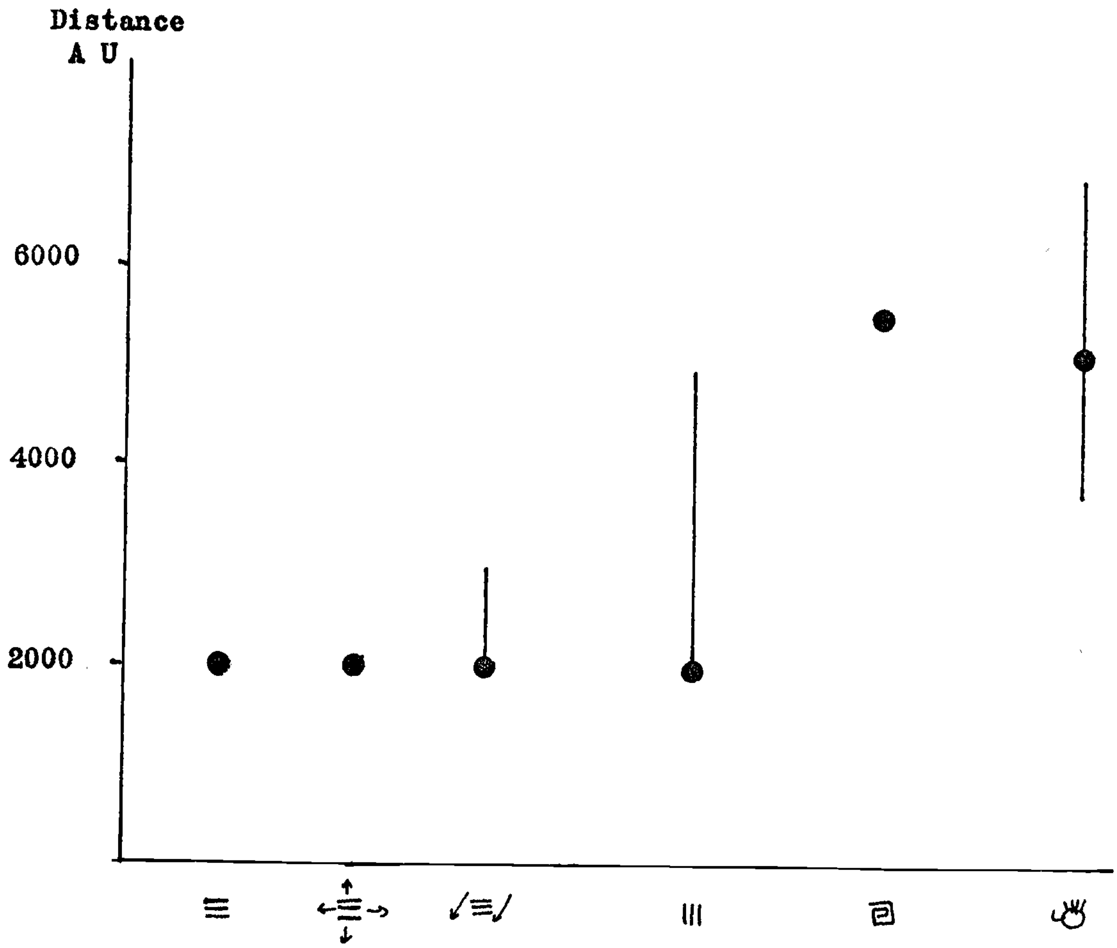
Symbol	Scan	Size	Position	Errors per alphabet
	XSCAN	60	Central	3
	XSCAN	80	Central	3
	XSCAN	60	Lower Left	1
	YSCAN	60	Central	6
	HATCH	60	Central	1
	SCRIBBLE	60	Central	17

Table 10.2

Trials of slices model



The letters were 60 or 80 A U high in a field 100 * 100

Figure 10.6

Results of slices simulation.

matching model and the hypothesis testing model. The accuracy is moderate in this form. This is a consequence of the simplicity of the program. At a cost of greater complexity, and thus loss of conceptual clarity, it could be improved. Human recognisers made some, if fewer, errors. The system adapts easily to various sizes and positions of letters. Its response to the different scans is similar to that of humans.

In all that it is successful. It is less successful in requiring twice as much scanning as people, though this is nearer than any other model achieved, and is due as much to inadequate programming as to conceptual error. It is a fair parallel of human recognition of letters in this task.

Discussion

The three recognition systems gave the results that might have been expected on theoretical grounds. The hypothesis testing model matched letters to "E" or "B" very efficiently, with some errors. The stencil matching model failed when it made an inaccurate estimate of the size of a letter. The slices model worked quite well, but not as well as humans fed on machine scan.

Thus it has been shown that hypothesis testing is possible, and very efficient, but not used by the subjects. Even those trials as true hypothesis testing were not solved as efficiently as by the hypothesis testing simulation. The difficulty of recognition by a stencil matching process implies that the task of recognising by scanning is much more difficult than simple recognition of letters. This makes the efficiency of the hypothesis testing model more remarkable.

Both the slices and the stencil matching recognition models have the characteristic of feature analysis that the recognition system either identifies the letter or reports it "unrecognisable". A feature analysis system gives no guidance to what information would assist recognition of something judged unrecognisable.

Note the internal representations of the letters in these models. The feature analysis models have representations which are geometric. The hypothesis testing representation is a set of conditions which must be true.

This distinction is convenient but not necessary. The hypothesis testing model can be arranged to deduce conditions and discriminations from stored shape information; sets of conditions of existence or non-existence of features are possible and natural representations for feature analysis.

The slices model produces distances to recognition relatively near those for human recognition of machine scans, and a similar ordering of efficiency of scans. This is a satisfactory result in a process designed to be an analogue of human processes, so it is a possible model of human letter recognition.

Other feature analysis models than slices could be equally efficient. Two of these are:

1. A model which analyses the delineation into a matrix of cells (say 5 x 7) and then analyses this as a set of points.
2. A set of probabilistic feature detectors working at various points in the delineation. Note that because this is feature analysis, the data about what features are obscure is not available to guide further scanning. This is similar to the analysis of leading and trailing edges in the slices model.

There is no a priori reason to think either of these models more efficient in recognition from delineations than the slices model.

These recognition models are virtually useless as machine recognisers in their present form. Even the hypothesis testing model took about three minutes to match a "B" against each letter of the alphabet, and the other models are slower. Most of this time was spent scanning the letter, and handling the list of points, which was often 1000 long.

As processes which could be mechanised to recognise normal print, rather than these scanned delineations, the utility of the systems is neatly reversed. The decisions that have to be made for stencil matching can be made in parallel, and the process can be mechanised to recognise letters in 10 micro seconds, if they are well defined and in the right place. The slices process would probably take longer to recognise one letter, but, at a cost in complexity, could be made self-adapting to different type-styles and sizes because most of the decisions it makes are relative, rather than absolute. The hypothesis testing model only matches letters, but could be extended to identify them. It intrinsically recognises by sequential processing, which makes it slow for simple recognition tasks. Its merit is that it needs less information (on average) for each decision, and so it is useful where information is expensive. One example of this kind of situation is in the recognition of nuclear particle tracks from bubble chamber photographs. There is so much information in a picture that it cannot all be taken into computer memory, and selective scanning must be used. One way of doing this is to hypothesise that a bright point is the start of a particle track, and scan along that track. Doing this for all bright points found by sparse scanning or human assistance means that all the tracks will be scanned, and that the blank areas will only be sparsely sampled, and not represented in the output data. (Brown, 1969)

This discussion has focussed on the potential of the three models for general letter recognition. However, the simulation focussed on the recognition of scanned delineations. For these, the hypothesis testing model shows how efficient hypothesis testing can be. The slices model shows one possible model of what people actually do, and the failure of the stencil-matching model points this up by showing one feature analysis process that does not work.

Errors

Subjects made errors in their identification in all the experiments. The total numbers were:

Experiment	Number of errors
Paradigm experiment	16
Constrained manual scans	7
All manual scans	9
Variable targets	14

When an erroneous judgement had been made, the trial was allowed to continue to a successful judgement, if appropriate, ie, in the three, four and five letter frames, and then included in the results. Others were discarded. No pattern in the errors was discernable.

The paucity of errors is surprising. Subjects were not told to hold to one confidence level, and those that inquired were told to choose one convenient to themselves, but to keep it constant.

Some subjects could not identify the targets. These trials were terminated after 600 seconds, and a positive result credited.

Chapter 11

CONCLUSIONS

The conclusion of the study is clear - people do not use hypothesis testing, even though it is much more efficient for this task. This is borne out by every experiment. Subjects do not use the scan to test hypotheses, but scan the whole letter or shape enough to make it recognisable by feature analysis. The basic form of the experiment shows this; various artefacts are ruled out by secondary experiments.

The paradigm experiment showed that few subjects were rated as using hypothesis testing. This rating is subjective, and objective support is necessary to make the conclusion secure. This is given by the machine scans, which took less distance than manual scans to render a letter legible in all the experiments. This efficiency is only predictable if the subject does not use hypothesis testing when scanning manually. More powerful evidence is that shapes were recognised as easily as letters, arguing that the familiarity of the picture had no effect on its recognition, and that identifying letters rather than matching them took no more scanning.

The reason that subjects used feature analysis is not that they were not able to control the joystick well enough to draw rectilinear lines, which might make for easier hypothesis testing, because when they were given a control system that made drawing rectilinear much easier than curved ones, in the Constrained Manual Scans Experiment, they did not benefit from it. Subjects use of feature analysis was not affected by the example of the machine scans; when no machine scans were seen by naive subjects their manual scanning was very similar to those subjects who saw manual scanning.

The constrained manual scan trials also show that increasing the difficulty of scanning, and thus its presumed cost, did not reduce the amount used, in contradiction to expectations. This indicates that increasing the cost of scanning over this range does not increase hypothesis testing, although greater increases might.

The explanation may be that the cost of thinking has also been raised; the subject must attend to the joystick, and not to optimising the scan path. This results in more scanning, and an even lower probability of hypothesis testing.

The manual scanning of letters is consistent with a scan mechanism that scans the field at random (no order was found in the recognition sequence of three letters), and then places a line alongside a line already found black. This rule would lead to lines parallel to one another and close together, which were found. The scan lines from manual trials are closer together than need be, as evidenced by the wider spacing of machine scans at recognition.

Also, the subjects in this experiment appeared to need enough scanning to make the connections between the scan lines interpretable by a simple rule - that nearest ends join together to form the edge of the letter. This only works if neighbouring scan lines are closer than the width of most parts of the picture. Thus this rule gives a scan density determined by the picture presented, and little affected by the recognition processes of the subject (the line joining process is seen as pre-processing).

Thus the subjects

1. scanned the field coarsely to find a black patch,
2. scanned near that black patch by parallelling an existing line until it became an identifiable letter,
3. stopped or repeated to find another letter.

This process was simulated in the slices model, and shown to agree moderately well with the human data.

The experiment was not designed to investigate feature analysis, but nevertheless some facts about the system are revealed. Letters were recognisable when they were 80% scanned, in general. Very few were less than 60% scanned, and complete scanning, meaning that no white gaps were left in black parts of the letter, was not necessary. The system requires a complete, but coarse view of the whole letter, without needing details of the corners or other micro-features. This is likely also to be the form of the internal representation. This is reasonable, as letters are commercially displayed as a 7 * 5 array of dots.

One other indicative phenomenon is that a diagonal scan is less efficient than a rectilinear scan. This must be because the information in diagonal scan lines is contradictory to the information in the letter, which is thus revealed to be stored in rectilinear form. This is similar to the Fraser spiral and letters. (Robinson, 1972)

This data agrees with other models of letter recognition based on psychological evidence.

Neisser (1967) worked with a list-searching paradigm, where subjects had to find a target letter among a list. He found that after some practice, subjects took about the same time per line for a variety of line-widths, and number of potential target letters. It was more difficult to find target letters among letters of similar shape than different shapes. He interprets this as the subject analysing each fixation by feature detectors, and, through training, being sensitive only to those features specifying target letters. This process is simulated in "Pandemonium", an archetypal feature analytic system.

However, for the more complex problem of recognising handwriting, he advances the notion of analysis by synthesis - archetypal hypothesis testing.

Gibson (1969) looked at the confusability of capital letters with children and adults, and showed that the confusions are compatible with a mechanism that detects features ("graphemes") in the letters, such as vertical lines, intersections of lines and symmetry. Letters differing in few features are more likely to be confused than those differing in many features. This is a more complex feature analytic process than pandemonium, and is a strong indication of the form of the internal representation.

Sternberg (1967) writes about two operations in character recognition, which correspond to the abstracting and comparing phases of feature analysis described before. He deduced from their reaction time in classifying digits as targets or non-targets, that subjects created an internal representation of the stimulus digit and then serially compared this with the representations of the target set. The effect of degrading the stimulus image is almost solely on the conversion of the image to internal representation. The effect of putting more digits in the target class is almost solely on the comparison of representations.

Some reasons for the lack of hypothesis testing in this task may be refuted.

1. The subject did not have enough time to employ a complex cognitive strategy.

   The trials took more than a minute, in general, and this should be long enough. If a longer time is necessary and sufficient then it seems unlikely to be fast enough for the real world.

2. These subjects were too stupid to use hypothesis testing.

   The subjects were students, in general, who were intelligent and flexible. It seems unlikely that others would succeed where these fail. However, it would be interesting to extend this work to young children, who are learning to recognise letters and simple shapes, and see if they use hypothesis testing. If so, it would support the theory that hypothesis testing is used to recognise novel patterns, and feature analysis for common ones.

3. Letters are exceptional, and always recognised by feature analysis.

   It is also necessary to argue that the results from shapes are artefactual, as these were recognised less by hypothesis testing than letters.

4. The processor necessary for hypothesis testing was fully occupied by scanning.

   However, machine scanning needs no effort from the subject, and produced no more partially scanned target letters than manual scan. They were more efficient than manual scans in distance per letter, but this was because the scan was more evenly spaced, not because fragments of a letter were adequare for decision. Also, operate a "scan then look" strategy, which means the processor is potentially available for hypothesis testing in the "look" phases.

5. The hypothesis can only be tested against the internal representation of the picture, and intermediate hypotheses are not available to drive the scan.
   This would certainly explain the results, and would be true if the internal representation was based on Fourier Transform. Hypothesis testing, under these conditions, is equivalent to feature analysis. A related possibility is that scanning with the hand cannot be controlled, but that scanning with the eyes could be. A Judas Eye driven by eye movements would test this possibility.

6. Hypothesis testing is not available.
   The existence of a few trials with definite hypothesis testing shows that humans can use it. Rosenberg (1971) set up a hypothesis testing number recognition program, working with analogues of biological line detector cells and a syntactic description of the pattern. This worked accurately. The skeletal hypothesis testing simulation of this study worked with less scanning and greater accuracy than either of the feature analysis models. Both of these computer models, and the human data, imply that hypothesis testing was possible.

Alternatively, hypothesis testing might not be available because the task was too simple, and that hypothesis testing is normally used to refine an original percept produced by feature analysis. Thus the blue Mini carrying a brass bedstead can be considered as the conjunction of many concepts, each with a relatively small range, such as blue (of anything), Mini (of cars). This organisation which could be feature analysed, and synthesised into a percept, is appealing, but it needs to be stretched to explain why the mechanism cannot match a letter in a complex display.

Thus hypothesis testing is either not used for such simple tasks (6), or in a form that does not reveal itself (5).

Many experiments in recognition measure the ease of recognition of a pattern by the amount of degradation the image can withstand while remaining recognisable. Techniques of doing this include tachistoscopes, image quantisation (Arps et al, 1969), deletion of the image (Gollin, 1965; Spitz & Borland, 1971; Williams, 1972). The line scanning technique adds another element to this armoury. However it seems dubiously usable as a general technique, and points up an objection in other models. The difficulty is that the part of the image that is given must be seen in its correct context - that the relation of parts of the image is as important as the parts themselves.

The argument is that deleting segments of a picture removes not only the information in the deleted segments, but also information about the connectivity of the remainder.

Two objections to the moving pen technique exist. This study has shown that hypothesis testing is not used for recognition of simple shapes, and the technique is not well adapted to the delineation of more complex shapes which might entail hypothesis testing. A Judas Eye driven by eye movements is the most promising technique for this.

Further, the moving pen technique is only revealing if hypothesis testing is operative; it does not distinguish between models of feature analysis.

In conclusion, these experiments were designed to investigate the hypothesis testing model of pattern recognition. Although hypothesis testing was efficient in the rare cases where the subject used it, it was not generally used. This may be because of the simplicity of the task, because the hypothesis cannot be used to operate a hand-controlled scan, or because hypothesis testing does not exist. Given that the task was designed to be best solved by hypothesis testing, that it is found so rarely is surprising if hypothesis testing is a normal mechanism of human pattern recognition.

REFERENCES

ARPS, R B, ERDMAN, R L, NEAL, A S, & SCHLAEPFER, C E. (1969) — IEEE Trans Man-Machine Systems MMS-10.3 66-71

ATTNEAVE, F. (1954) — Some Informational Aspects of Visual Perception. Psychological Review 61 183-193

BLAKEMORE, C. (1973) — The baffled brain. In - Illusion in Nature and Art. Eds. GREGORY, R L & GOMBRICH, E H. Duckworth (London).

BLOOMFIELD, J R (1970) — Visual Search. PhD Thesis, University of Nottingham.

BORNAT, R & BRADY M (1974) — The Essex University Vision Project. - Proposal.

BRITISH COMPUTER SOCIETY (1967) — Character Recognition. London.

BROWN, R M (1969) — Pattern Recognition problems in bubble chamber physics. Pattern Recognition 1 263-271.

BURSTALL, R M, COLLINS, J S, & POPPLESTONE, R J (1968) — The POP-2 Papers. Edinburgh University Press.

DAVIDSON, M L (1968) — Perturbation approach to spatial brightness interaction in human vision. Journal of the Optical Society of America. 58 1300-1309.

DEREGOWSKI, J B, MULDROW, E S, & MULDROW, W F (1972) — Pictorial recognition in a remote Ethiopian population. Perception 1 417-426.

EMPODOCLES (c. 440 BC) — Reported in Encyclopaedia Brittannica (1963 Edition). Vol 18 page 709, London.

GARDNER, M (1961) — More Mathematical Puzzles and Diversions from Scientific American. pages 130-137. Penguin. Harmondsworth (1966)

GIBSON, E J (1969) — Principles of Perceptual Learning and Development. Appleton Century Crofts. New York.

GIBSON, J J (1966) — The Senses Considered as Perceptual Systems. George Allen and Unwin. London.

GOLLIN, E S (1965) — Perceptual learning of incomplete pictures.
Perceptual and Motor Skills
21 439-445.

GREGORY, R L (1966) — Eye and Brain.
Weidenfeld (London).

GREGORY, R L (1970) — The Intelligent Eye.
Weidenfeld (London).

GREGORY, R L (1973) — The Confounded Eye
In - Illusion in Nature & Art
Eds GREGORY, R L & GOMBRICH, E H.
Duckworth (London).

GREENWOOD, E (1973) — Reconnaisance and arms control.
Scientific American 228.2 14-25.

GROSS, C G, ROCHA-MIRANDA, C E, BENDER, D B (1972) — Visual properties of neurons in inferotemporal Cortex of the macaque.
J. Neurophysiol. 35 96-111.

GUZMAN, A (1968) — Decomposition of a visual scene into three-dimensional bodies.
AFIPS Proceedings of the Fall Joint Computer Conference.
33 291-304.

HACKMAN, R B & GUILFORD, J P. (1936) — A Study of the Visual Fixation Method of Measuring Attention Value.
J. Appl. Psychol. 20 44-59.

HOCHBERG, J & BROOKS, V (1962) — Pictorial recognition as an unlearned ability: a study of one child's performance.
American Journal of Psychology
75 624-628.

HOCHBERG, J (1968) — In the Mind's Eye. In - Haber, R N (Ed) Contemporary Theory and Research in Visual Perception.
Holt, Rinehart & Winston. London.

HOWE, J A M (1965) — The Role of Eye Movements in Visual Perception and Recognition.
PhD Thesis. University of Cambridge.

HUBEL, D H & WIESEL, T N (1962) — Receptive Fields, Binocular Interaction and Functional Architecture in Cat's Visual Cortex.
J. Physiol. 160 106-154.

KENNEDY, J M (1974) A psychology of picture perception. Jossey-Bass (San Francisco).

KÖHLER, W (1963) Gestalt Psychology. In - Encyclopaedia Britannica (London) 1963 Edition. Vol 18 pp 715-717.

KORN, G A & KORN, T M (1964) Electronic Analogue and Hybrid Computers. McGraw-Hill. New York.

LEE, D N (1974) Visual Information during locomotion. In - MACLEOD, R B & PICK, H L (Eds) Perception: Essays in honour of James J Gibson. Cornell University Press. Ithaca.

LEONARD, J A & NEWMAN, R C (1970) Three Types of Visual Map for Blind Travel. Ergonomics <u>13</u> (2) 165-179.

LETTVIN, J Y, MATURANA, H R, McCULLOCH, W S & PITTS, W H (1959) What the Frog's Eye Tells the Frog's Brain. Proc. I R E <u>47</u> 1940-1951.

LOOMIS, J M (1974) Tactile letter recognition under different modes of stimulus presentation. Perception & Psychophysics <u>16.2</u> 401-408.

MACKWORTH, N (1965) Visual Noise Causes Tunnel Vision. Psychonomic Science <u>3</u> 67-68.

MACKWORTH, N & SCHISSLER, D R (1966) Analysis of pictures by subjective Ratings and by eye Fixations. Interval Report, Harvard Centre for Cognitive Studies, October 1966.

MORRIS, A E & HORNE, E P (Eds) (1960) Visual Search Techniques. National Academy of Sciences - National Research Council Publication 712. Washington.

MURPHY, A S (1969) An application of Heuristic Search Procedures to Picture Interpretation. MIP-R-61. Department of Machine Intelligence and Perception. Edinburgh.

NEISSER, U (1967) Cognitive Psychology. Appleton Century Crofts. New York.

NOTON, D & STARK, L (1971) — Scanpaths in Eye Movements during Pattern Perception. Science 171 308-311.

PLATO (c. 350 BC) — Reported in Encyclopaedia Britannica (London) Vol 18 p 709 & p 60.

PODD'YAKOV, N N (1970) — On the Origins of Recognition Processes in the Visual System. Voprosy Psikhologii 16 139-141.

ROBINSON, J O (1972) — The Psychology of Visual Illusion. Hutchison University Library. London.

ROSENBERG, B (1971) — Character Recognition & Visual Processing. Int. J. Man-Machine Studies 3 189-200.

SAKAI, T, NAGAO, M & FUJIBAYASHI, S (1969) — Line extraction and pattern detection in a photograph. Pattern Recognition 1 233-248.

SELFRDIGE, O & NEISSER, U (1960) — Pattern Recognition by Machine. Scientific American.

SOKOLOV, E N (1970) — The Modelling Properties of the Nervous system. In - Cole & Maltzman (Eds) Contemporary Soviet Psychology. Plenum Press. New York.

SPITZ, H H & BORLAND, M D. (1971) — Redundancy in line drawings of familiar objects: effects of age and intelligence. Cognitive Psychology 2 196-205.

STERNBERG, S (1967) — Two Operations in Character Recognition: Some Evidence from Reaction Time Measurements. In - Wathen-Dunn (Ed), Models for Perception of Speech and Visual Form. MIT Press. Cambridge, Mass.

TIKHOMIROV, O K & POZNYANSKAYA, E D (1966) — An Investigation of Visual Search as a Means of Analysing Heuristics. Voprosy Psikhologii 12 (4) 39-53.

UHR, L & VOSSLER, C (1961) — A pattern recognition program that evaluates and adjusts its own operators. Proceedings of the Western Joint Computer Conference 19 555-569.

ULLMAN, J R & KIDD, P A (1969) Recognition Experiments with Typed Numerals in the Mail. Pattern Recognition 1 (4) 273-289.

WILLIAMS, R (1972) A new apparatus for visual data sampling. Perception 1 359-362.

WINSTON, P H (1972) The MIT Robot. In - Machine Intelligence 7. Eds. Meltzer, B & Michie, D. Edinburgh University Press.

YARBUS, A L (1967) Eye Movements and Vision. Plenum Press, New York.

DATA SYSTEM

((LETTERS))

2. SQ — A squared.

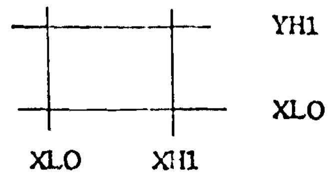
4. DIST — Distance of point (A, B) from (C, D).

9. DQ — Perpendicular distance of point (X1,X1) from a line through (X2, Y2) & (X3, Y3).

This is positive on the right of the line and negative on the left.

13. CIRCLE — This is true if (X, Y) is within a circle of radius RAD and centre (XCENTRE, YCENTRE).

16. SQUARE — True if (X, Y) is within a rectangle.



20. CEE — True if (X, Y) is between two circles with a common centre and to the left of a line.

26. BAR — True if (X, Y) is within the quadrilateral with corners (X1, Y1) .. (X4, Y4).

37. MACRO XY — Saves writing (X-XS), (Y-YS) each time. (XS, YS) is start position (bottom left corner).

39 - 41 — Definition of A, based on a height of 50.

42 - 130 — Similarly for the other 25 letters.

[LETTERS] [23.25 27 5 75]

```
FUNCTION SQ A; A*A; END;

FUNCTION DIST A B C D;
SQRT(SQ(A-C)+SQ(B-D));END;



FUNCTION DD X1 Y1 X2 Y2 X3 Y3;
((Y1-Y2)*(X1-X3)-(Y1-Y3)*(X1-X2))/DIST(X1,Y1,X2,Y2);
END;

FUNCTION CIRCLE X Y XCENTRE YCENTRE RAD;
(DIST(X,Y,XCENTRE,YCENTRE)<RAD); END;

FUNCTION SQUARE X Y XHI XLO YHI YLO;
IF (X>XLO) AND (X<XHI) AND (Y>YLO) AND (Y<YHI)
THEN TRUE ELSE FALSE CLOSE; END;

FUNCTION CEE X Y XCEN YCEN RADHI RADLO X1 Y1 X2 Y2;
IF CIRCLE(X,Y,XCEN,YCEN,RADHI) AND
NOT(CIRCLE(X,Y,XCEN,YCEN,RADLO)) AND
(DD(X1,Y1,X2,Y2,X,Y)<0)
THEN TRUE ELSE FALSE CLOSE ; END;

FUNCTION BAR X Y X1 Y1 X2 Y2 X3 Y3 X4 Y4;
IF (DD(X1,Y1,X2,Y2,X,Y)>0) AND
(DD(X2,Y2,X3,Y3,X,Y)>0) AND
(DD(X3,Y3,X4,Y4,X,Y)>0) AND
(DD(X4,Y4,X1,Y1,X,Y)>0)
THEN TRUE ELSE FALSE CLOSE ;END;


VARS A00 B00 C00 D00 E00 F00 G00 H00 I00 J00 K00 L00
M00 N00 O00 P00 Q00 R00 S00 T00 U00 V00 W00 X00 Y00 Z00;

MACRO XY; [(X-XS),(Y-YS)]; .MACRESULTS;END;

[BAR(XY,0,0,20,50,20,30,10,0) OR
BAR(XY,20,50,40,0,30,0,20,30) OR
SQUARE(XY,30,10,13,7)]->A00;

[SQUARE(XY,8,0,50,0) OR
CEE(XY,15,33,12,5,8,50,8,0) OR
CEE(XY,15,15,15,7,8,50,8,0) ]->B00;

[CEE(XY,25,25,25,16,35,0,35,50)]->C00;

[SQUARE(XY,8,0,50,0) OR
CEE(XY,15,25,25,16,8,50,8,0)]->D00;

[SQUARE(XY,8,0,50,0) OR
SQUARE(XY,38,8,8,0) OR
SQUARE(XY,38,8,50,42) OR
SQUARE(XY,33,8,28,20)]->E00;

[SQUARE(XY,8,0,50,0) OR
```

```
SQUARE(XY,38,8,50,42) OR
SQUARE(XY,33,8,28,20)]->F00;

[CEE(XY,25,25,25,16,40,0,40,50) OR
SQUARE(XY,40,32,15,0)]->G00;

[SQUARE(XY,8,0,50,0) OR
SQUARE(XY,40,32,50,0) OR
SQUARE(XY,32,8,28,20)]->H00;

[SQUARE(XY,8,0,50,0)]->I00;

[CEE(XY,15,15,15,7,40,15,0,15) OR
SQUARE(XY,30,22,50,15)]->J00;

[SQUARE(XY,8,0,50,0) OR
BAR(XY,0,35,20,50,30,50,8,25) OR
BAR(XY,5,35,30,0,20,0,8,20)]->K00;



[SQUARE(XY,8,0,50,0) OR
SQUARE(XY,30,8,8,0)]->L00;

[SQUARE(XY,8,0,50,0) OR
BAR(XY,8,50,30,0,20,0,8,35) OR
BAR(XY,42,50,42,35,30,0,20,0) OR
SQUARE(XY,50,42,50,0)]->M00;

[SQUARE(XY,8,0,50,0) OR
BAR(XY,8,50,32,15,32,0,8,35) OR
SQUARE(XY,40,32,50,0)]->N00;

[CEE(XY,25,25,25,17,50,0,60,50)]->O00;

[SQUARE(XY,8,0,50,0) OR
CEE(XY,15,35,15,7,8,50,8,0)]->P00;

[CEE(XY,25,25,25,17,60,0,60,50) OR
BAR(XY,40,20,50,0,40,0,30,20)]->Q00;

[SQUARE(XY,8,0,50,0) OR
CEE(XY,15,35,15,7,8,50,8,0) OR
BAR(XY,18,25,30,0,20,0,8,25)]->R00;

[CEE(XY,15,35,15,7,0,0,30,50) OR
CEE(XY,15,15,15,7,30,50,0,0)]->S00;

[SQUARE(XY,34,0,50,42) OR
SQUARE(XY,21,13,50,0)]->T00;

[CEE(XY,15,15,15,7,30,15,0,15) OR
SQUARE(XY,8,0,50,15) OR
SQUARE(XY,30,22,50,15)]->U00;

[BAR(XY,0,50,10,50,20,20,20,0) OR
BAR(XY,40,50,20,0,20,20,30,50)]->V00;

[BAR(XY,0,50,10,50,20,0,10,0) OR
BAR(XY,15,50,20,0,10,0,25,50) OR
```

132.	LETTER	LETTER (letter , start position, size) gives a function (say BFN), where BFN (X, Y) is true if X, Y is black in that letter.
134.		Rejects the point if it is outside the area of the letter. This speeds things up.
138.		Adjusts the letter to the required size.
141.		Adjusts the letter to the required start.
144.	SHOWL	Prints the whole letter, to allow its shape to be seen.
158.	SHOWALL	PRINTS all 26 letters.

```
BAR(XY,25,50,35,50,50,0,40,0) OR
BAR(XY,50,50,50,0,40,0,50,50)]->W00;

[BAR(XY,0,50,10,50,35,0,25,0) OR
BAR(XY,25,50,35,50,10,0,0,0)]->X00;

[BAR(XY,0,50,10,50,20,30,20,15) OR
BAR(XY,40,50,20,15,20,30,30,50) OR
SQUARE(XY,24,16,30,0)]->Y00;

[SQUARE(XY,32,0,50,42) OR
SQUARE(XY,35,3,8,0) OR
BAR(XY,25,50,35,50,10,0,0,0)]->Z00;

FUNCTION LETTER L XS YS SIZE;
[LAMBDA X Y XS YS;
IF SQUARE(X,Y,XS+2+]<>[%SIZE*1.25%]<>[,XS-2,YS+5+]<>[%SIZE%]<>[,YS-2)
THEN ELSE FALSE EXIT;
 IF]<>
(MAPLIST((L,1.CHARWORD;0,0,3.CONSWORD.VALOF),
 LAMBDA S; IF S.ISNUMBER THEN S*SIZE/50 ELSE S; CLOSE;
 END;);)<>
[THEN TRUE ELSE FALSE CLOSE; END; GOON;];.POPVAL;
(%XS,YS%); END;


FUNCTION SHOWL NO;
VARS X Y M BFN;
LETTER(CONSWORD(NO,1),5,5,49)->BFN;
NEWANYARRAY([0 70 0 60], LAMBDA I J ;
IF BFN(I,J) THEN NO ELSEIF (I//10;.ERASE)=0 OR
(J//10;.ERASE)=0 THEN 30 ELSE 16; CLOSE; END;
INITC,SUBSCRC)->M;
64.CUCHAROUT; 60->Y; 0->X;
LY:
M(X,Y).CUCHAROUT;
X+1->X; IF X<71 THEN GOTO LY; CLOSE;
Y-1->Y;0->X; 1.NL; IF Y THEN GOTO LY; CLOSE;
END;

FUNCTION SHOWALL N;

L: N.SHOWL; N+1->N; IF N<59 THEN GOTO L CLOSE;
END;
```

	MOVES	
5.	RESET	Sets up starting conditions.
11.	MOVE	The basic scanning function. It scans from the present position to (X, Y). At unit intervals it samples the letter; if black, that point is added to POINTLIST. It also counts writing distance DISTB and CROSSINGS.
32.	RANDMOVE	Moves to a random point.
37.	SCRIBBLE	Moves to Q random points in succession.
40.		This makes sure the point scanned to is outside the boundaries of the letter, which has been found by a recognition attempt. If this is not done, too much of the scanning happens in the middle of the letter, and the periphery is badly defined.
43.	XSCAN	TV(X) simulation. STEP is the separation lines, N is the number of lines to be scanned.
50.	YSCAN	TV(Y) simulation.
56.	MATCH	Square spiral simulation. INGOING is true if the spiral is contracting, false otherwise.

```
VARS XNOW YNOW ENDSLIST POINTLIST DISTANCE DISTB CROSSINGS
XMIN YMIN XMAX YMAX ;
FUNCTION RESET;50->XNOW; 50->YNOW;
[]->ENDSLIST; []->POINTLIST;
0->DISTANCE; 0->DISTB; 0->CROSSINGS;
END;
.RESET;
FUNCTION MOVE X Y ;
VARS D STEP XSTEP YSTEP B BWAS ;
BEN(XNOW,YNOW)->BWAS; DIST(X,Y,XNOW,YNOW)->D ;
IF D=0 THEN EXIT;D.INTOF->STEP;
(X-XNOW)/D->XSTEP; (Y-YNOW)/D->YSTEP;
DISTANCE+D->DISTANCE;
L1:
XNOW+XSTEP->XNOW; YNOW+YSTEP->YNOW;
BEN(XNOW,YNOW)->B;
IF B=BWAS THEN ELSE CROSSINGS+1->CROSSINGS;
 [%(XNOW-(XSTEP/2)),(YNOW-(YSTEP/2))%]::ENDSLIST->ENDSLIST;
CLOSE;
IF B THEN DISTB+1->DISTB;[% XNOW,YNOW %]::POINTLIST->POINTLIST;
CLOSE;
B->BWAS; STEP-1->STEP;
IF STEP THEN GOTO L1; CLOSE;
END;
[RANDPACK].LIBRARY.COMPILE;
4321->RANSEED;
FUNCTION RANDMOVE;
MOVE(RANDREAL(100),RANDREAL(100)); END;
100->XMIN; 100->YMIN;
0->XMAX; 0->YMAX;
 FUNCTION SCRIBBLE Q;
VARS X Y ;
L1:  100.RANDREAL->X; 100.RANDREAL->Y;
IF SQUARE(X,Y,XMAX,XMIN,YMAX,YMIN,) THEN GOTO L1; CLOSE;
MOVE(X,Y); Q-1->Q;
IF Q THEN GOTO L1; CLOSE; END;
FUNCTION XSCAN STEP N;
L1:0->XNOW; YNOW-STEP->YNOW;
IF YNOW=<0 THEN YNOW+100->YNOW; CLOSE;
MOVE(100,YNOW); N-1->N;
IF N THEN GOTO L1; CLOSE;
END;
FUNCTION YSCAN STEP N;
L1: 100->YNOW; XNOW-STEP->XNOW;
IF XNOW=<0 THEN XNOW+100->XNOW; CLOSE;
MOVE(XNOW,0); N-1->N;
IF N THEN GOTO L1; CLOSE; END;
VARS INGOING ; 1->INGOING;
FUNCTION HATCH STEP CYCLES;
IF XNOW<50 THEN 100-XNOW->XNOW; CLOSE;
```

```
L1: XNOW->YNOW;
 MOVE(XNOW,100-YNOW);
 MOVE(100-XNOW,YNOW);
 MOVE(XNOW,100-YNOW);
 MOVE(100-XNOW,YNOW);
IF INGOING THEN XNOW-STEP ELSE XNOW+STEP;CLOSE->XNOW;
L3:IF XNOW>100 THEN 200 -  XNOW->XNOW; 1->INGOING; CLOSE;
L2: IF XNOW<51 THEN 100- XNOW->XNOW;0->INGOING; CLOSE;
CUCHAROUT; CHAROUT->CUCHAROUT; XNOW.PR; ->CUCHAROUT;
CYCLES-1->CYCLES; IF CYCLES THEN GOTO L1; CLOSE; END;
```

DISPLAY

3. SLICEPR	Displays the slices and edges. The original data is local to a function inside INSPECT, so it is recomputed here.
16. DEPOSE	Prints to the delineation CHAROUT Prints on the teletype, CUCHAROUT on the line printer.
30.	Set up a letter.
33.	Scan it and try to recognise it.
34.	IF not recognised, and not over distance, try again.
36 - 38.	Print the data.
39, 40.	Go on to next letter.

```
FUNCTION SLICEPR;
VARS SLICE QL;
SLICENO.GETSLICES; .LIMITS;
APPLIST(POINTLIST,FILLSLICE(%SLICENO%));
1,SLICE.FROZVAL.DATALIST.REV->QL;
APPLIST(QL,LAMBDA S;2.NL; S.COLUMNISE.PRSTRING;END;);
3.NL;
APPLIST(QL,LAMBDA S;1.NL; S.ENCODE.PR;END;);
.LIMFIND; "EDGEF"=>;EDGEF.REV.PR;
"EDGEB"=>;EDGEB.REV.PR;
END;


FUNCTION DEPOSE;
VARS M X Y N P ;
.LIMITS;
NEWANYARRAY([0 50 0 50],LAMBDA I J; IF (I//10;.ERASE)=0
OR (J//10;.ERASE)=0 THEN 30 ELSE 16 CLOSE; END;,INITC, SUBSCRC)->M;
SLICENO+1->N;
LOOPIF N THEN N-1->N; INTOF((YMIN+N*HEIGHT/SLICENO)/2)->P;
  50->X; LOOPIF X THEN X-1->X; 29->M(X,P); CLOSE;
CLOSE;;
APPLIST(POINTLIST,LAMBDA S; S.DEST.DEST.ERASE->Y->X;
 26->M(INTOF(X/2),INTOF(Y/2)); END;);
APPLIST(ENDSLIST, LAMBDA S; S.DEST.DEST.ERASE->Y->X;
 56->M(INTOF(X/2),INTOF(Y/2)); END;);
50->Y; 0->X; 64.CUCHAROUT;
L1:
M(X,Y).CUCHAROUT;
X+1->X; IF X=<50 THEN GOTO L1; CLOSE;
Y-1->Y; 0->X; 1.NL; IF Y THEN GOTO L1; CLOSE;
2.NL;
'DISTANCE = '.PRSTRING; DISTANCE.PR; 1.NL;
'WRITING DISTANCE = '.PRSTRING; DISTB .PR; 1.NL;
'CROSSINGS = '.PRSTRING; CROSSINGS.PR; 1.NL;
END;
```

Supervisory functions for "slices" model

5.	TRY	A supervisory function, superseded.
22.	DUMP	Prints L to dise.
26.	ATTEMPT	Sets up and attempts to recognise each letter. The normal supervisory function for the "slices" model.

```
VARS TITLE DUMP SCAN;

FUNCTION TRY;
33->N;
LP:
CONSWORD(N,1)->P;
LETTER(P,3,3,73)->BFN;
.RESET;
P=>;1.NL;
XSCAN(1,100);
.DEPOSE;
1.NL; POINTLIST.INSPECT.PR;
1.NL;
UNKNOWN.PR;
N+1->N;
GOTO LP;
END;


FUNCTION DUMP L;
CUCHAROUT; DRAIN->CUCHAROUT;1.NL;
L.PR; 2.SP; ->CUCHAROUT; END;

FUNCTION ATTEMPT FIRSTLET SCAN TITLE XLO YLO SIZE;
VARS N P Q ;
FIRSTLET->N;
LP1: 17.CHAROUT;
1.CHAROUT; LETTER(CONSWORD(N,0,0,3),XLO,YLO,SIZE)->BFN;
.RESET;
LP3:
.SCAN; POINTLIST.INSPECT->Q; 13.CHAROUT;
IF Q.NULL AND DISTANCE<7000 THEN
GOTO LP3; CLOSE;
.DEPOSE; Q=>;UNKNOWN=>;.SLICEPR;
L% N,1.CONSWORD,Q,UNKNOWN,DISTANCE,DISTB,CROSSINGS,TITLE,
SCAN.ENPART,XLO,YLO,SIZE%].DUMP;
N+1->N;
IF N>58 THEN EXIT; GOTO LP1; END;
```

DICT

The dictionary for the slices recognition mechanism.

The layout is

Letter name, shape of leading (left) edge top, bottom, shape of trailing edge top, bottom arrangement of slices starting from the top .

The shapes of the edges can be

A	
V	like themselves, leading or trailing.
I	
S	Odd.

[DICT]   [23.26 27 5 75]

```
VARS DICT;
[
[A A A A A  [1 2 L 2 ]]
[B I I S S  [L 2 L 2 L ]]
[C A V V A  [L 2 1 2 L ]]
[D I I A V  [L 2 L]]
[E I I S S  [L 1 L 1 L]]
[F I I S V  [L 1 L 1]]
[G A V V A  [ L 2 1 2 L ]]
[H I I I I  [2 L 2]]
[I I I I I  [1]]
[J I S I V  [1 2 L ]]
[K I I S A  [ 2  1  2]]
[L I I I A  [1 L ]]
[M I I I I  [2 4 3 ]]
[N I I I I  [2 3 2 ]]
[O A V A V  [L 2 L ]]
[P I I S V  [L 2 L 1]]
[Q A V A S  [L 2 L ]]
[R I I S S  [L 2 L 2 ]]
[S S S S S  [L 2 L 2 L ]]
[T V I V I  [ L 1 ]]
[U I V I V  [2 L ]]
[V V V V V  [ 2 1 ]]
[W V V V V  [3 2 ]]
[X V A V A  [2 L 1 L 2]]
[Y V V V V  [2 1 ]]
[Z S A V S  [ L 1 L ]]
 ]->DICT;
```

SLICE

The second feature recognition system

6.		SLICENO	is the number of slices into which the letter is divided.
		MINL	is the minimum % of a letter width that consitutes a long line.
		MINGAP	is the maximum gap between points as % of letter width, which can be ignored.
9.	LIMITLOOK		S is a point that is a list X, Y . This increases the limits of the letter to include S.
18.	LIMITS		Finds the size and position of the letter revealed.
27.	SET SLICES		Sets up storage for the slices.
31.	FILL SLICES		Puts a point into its correct slice.
39.	APPINPAIRS		Applies FN to each successive pair of list L.
45.	FIRST		Produces a list of the first N items in LIST.
50.	LIMSEEK		Finds the biggest and smallest values in list L.
56.	LIMFIND		Finds the ends of each slice, and puts them in EDGEF (front edge) and EDGEB (back edge).

```



VARS XMAX XMIN YMAX YMIN HEIGHT WIDTH S L SLICE SLICENO
UNKNOWN MINL MINGAP EDGEF EDGEB NOUGHTS ONES PLIST;
9->SLICENO;
42->MINL; 5->MINGAP;

FUNCTION LIMITLOOK S;
VARS X Y ;
S.DEST.DEST.ERASE->Y->X;
IF X>XMAX THEN X->XMAX; CLOSE;
IF X<XMIN THEN X->XMIN; CLOSE;
IF Y>YMAX THEN Y->YMAX; CLOSE;
IF Y<YMIN THEN Y->YMIN; CLOSE;
END;

FUNCTION LIMITS;
0->XMAX; 0->YMAX;.
100->XMIN; 100->YMIN;
APPLIST(ENDSLIST,LIMITLOOK);
APPLIST(POINTLIST,LIMITLOOK);
YMAX-YMIN->HEIGHT;IF HEIGHT<1 THEN 1->HEIGHT; CLOSE;
XMAX-XMIN->WIDTH;IF WIDTH<1 THEN 1->WIDTH; CLOSE;
END;

FUNCTION SETSLICES N;SUBSCR(% INIT(N)%)->SLICE;
L: []->SLICE(N); N-1->N; IF N THEN GOTO L; CLOSE;
END;

FUNCTION FILLSLICE S N;VARS X Y H ;
S.DEST.DEST.ERASE->Y->X;
INTOF(N*(Y-YMIN)/HEIGHT)->H;
H+1->H; IF H>N THEN N->H; CLOSE;
X::SLICE(H)->SLICE(H);
END;


FUNCTION APPINPAIRS L FN;
VARS S ;
LL: L.DEST->L->S;
IF L.NULL THEN 1 EXIT;
IF FN(S,L.HD) THEN GOTO LL; ELSE 0 CLOSE; END;

FUNCTION FIRST N LIST;
VARS L; NIL->L;
LOOPIF N THEN LIST.DEST->LIST; ::L->L; N-1->N; CLOSE;
L.REV; END;

FUNCTION LIMSEEK L;
VARS XMIN XMAX; 1000->XMIN; 0->XMAX;
APPLIST(L, LAMBDA S; IF S>XMAX THEN S->XMAX; CLOSE;
 IF S<XMIN THEN S->XMIN; CLOSE; END;) ;
XMAX; XMIN; END;

FUNCTION LIMFIND ;
VARS N; SLICENO->N; []->EDGEF; []->EDGEB;
```

62.	EDGESORT	Finds the shape of the edge. EDGE is a list of the ends of each slice.
82.	COLUMNISE	Puts the points within a slice into their position in a strip of length 10 across the width of a letter.
91.	POSTGAP	Checks for short or long bars.
95.	ENCODE	LIST is a list of points in a slice, and the result is the type of lines in the slice, either the number of short segments or "L" for long segment.
112.	LINEFIND	1. If too many small line segments are found, (4) this increases MINGAP until they go away. 2. This removes one of identical neighbours from the list of sector types until each item is different from its neighbours.

```
LOOPIF N THEN N.SLICE.LIMSEEK;N-1->N;
::EDGEF->EDGEF; ::EDGEB->EDGEB; CLOSE;
END;

FUNCTION EDGESORT EDGE;
VARS TOP BOTTOM HALFHEIGHT;
 FUNCTION STRAIGHT EDGE;
 VARS XMIN XMAX; EDGE.LIMSEEK->XMIN->XMAX;
 IF (XMAX-XMIN)=<HEIGHT/50 THEN 1 ELSE 0 CLOSE;
 END;
IF EDGE.STRAIGHT THEN "I","I"; EXIT;
INTOF((SLICENO+1)/2)->HALFHEIGHT;
FIRST(HALFHEIGHT,EDGE)->TOP;
IF TOP.STRAIGHT THEN "I"
ELSEIF APPINPAIRS(TOP,NONOP=<) THEN "V"
ELSEIF APPINPAIRS(TOP,NONOP>=) THEN "A"
ELSE "S" CLOSE;
FIRST(HALFHEIGHT,EDGE.REV).REV->BOTTOM;
IF BOTTOM.STRAIGHT THEN "I"
ELSEIF APPINPAIRS(BOTTOM,NONOP=<) THEN "V"
ELSEIF APPINPAIRS(BOTTOM,NONOP>=) THEN "A"
ELSE "S" CLOSE;
END;

FUNCTION COLUMNISE LIST;
VARS CONST COLUMN N;
SUBSCRC(%INITC(101)%)->COLUMN;
 100/(IF WIDTH>HEIGHT THEN WIDTH ELSE  HEIGHT CLOSE;)->CONST;
101->N;
LOOPIF N THEN 0->COLUMN(N);N-1->N; CLOSE;
APPLIST(LIST, LAMBDA S; 1->COLUMN(INTOF((S-XMIN)*CONST+1));END;);
FROZVAL(1,COLUMN); END;

 FUNCTION POSTGAP;
 IF ONES>=MINL THEN "L" ELSE "S" CLOSE; ::PLIST->PLIST;
 0->ONES; 0->NOUGHTS; END;

FUNCTION ENCODE LIST;
VARS PLIST N ONES NOUGHTS T COLUMN;
0->ONES; 0->NOUGHTS;
1->N; []->PLIST;
SUBSCRC(%LIST.COLUMNISE%)->COLUMN;
L3:
IF COLUMN(N)=0 THEN 1+NOUGHTS->NOUGHTS;
 ELSE 1+ONES->ONES; CLOSE;
IF COLUMN(N)=0 AND COLUMN(N+1)=1 THEN
IF ONES THEN
 IF NOUGHTS>=MINGAP THEN .POSTGAP;
  ELSE NOUGHTS+ONES->ONES; CLOSE; CLOSE;0->NOUGHTS; CLOSE;
IF N<100 THEN N+1->N; GOTO L3; CLOSE;
IF ONES THEN .POSTGAP; CLOSE;0->T;
APPLIST(PLIST, LAMBDA S; IF S="L" THEN 1->T; CLOSE; END;);
IF T THEN "L" ELSE PLIST.LENGTH; CLOSE; END;

FUNCTION LINEFIND L;
VARS R Q KM KMM ; []->R;MINGAP->KMM;
L4:MAPLIST(L,ENCODE)->Q; MINGAP->KM;
APPLIST(Q,LAMBDA S; IF S.ISNUMBER AND S>4 THEN
 MINGAP+5->MINGAP; CLOSE; END;);
IF KM=MINGAP THEN ELSE GOTO L4; CLOSE;
```

123.	COMPARE		True if the two dictionary-entry-type items are identical.
			<u>The recognising function</u>
128.	INSPECT		LIST is usually POINTLIST.
129.		LIMITS	Find the apparent size & position of the letter.
130, 131.			Set up the slices, and put each point into the right one.
132.			Find the back and front edges.
134, 135.			Make a dictionary entry, name "UNKNOWN", from the data from pointlist.
136, 137.			Compare this with each dictionary entry, making a list of true results.

```
LQ: Q.DEST->Q->S;KMM->MINGAP;
IF S=Q.HD THEN ELSE S::R->R; CLOSE;
IF Q.TL.NULL THEN ((Q.HD)::R); ELSE GOTO LQ; CLOSE;
END

FUNCTION COMPARE DICT1 DICT2;
DICT1.TL.REV->DICT1; DICT2.TL.REV->DICT2;
IF EQUAL(DICT1.HD ,DICT2.HD) AND EQUAL(DICT1.TL, DICT2.TL)
THEN 1 ELSE 0 CLOSE; END;

FUNCTION INSPECT LIST;
.LIMITS;
SLICENO.SETSLICES;
 APPLIST(LIST, FILLSLICE(%SLICENO%));
.LIMFIND;
[% "UNKNOWN",EDGEF.REV.EDGESORT,
 MAPLIST( EDGEB.REV, LAMBDA S;111-S;END;).EDGESORT,
 1,SLICE.FROZVAL.DATALIST.LINEFIND. %]->UNKNOWN;
[% ( APPLIST(DICT, LAMBDA S; IF COMPARE(S,UNKNOWN) THEN
  S.HD; CLOSE; END;)) %];
END;
```

3.	MOVEFN	Scans from present position in a given direction until the edge of the picture is met or a boundary crossed. BW is 1 for black, 0 for white. The truth values of the function are back to front to make later programming neater. MOVEFN returns true if 1. It starts in the other colour from BW 2. It goes off the picture, ie, it fails false if 1. It starts in the right colour and then crosses a boundary, ie, it works.
14 - 18	MOVEUP, -DOWN, -LEFT, -RIGHT	are MOVEFN in these directions.
20.	STARTSCAN	Moves until something black is found.
28.	EGREG1	One hypothesis testing function. The picture is hit on its left hand edge.
30.		Move into the letter 1 unit.
31.		Scan to the top, return to within the letter.
33.		Scan to the bottom. This establishes the height. If the front is not straight, this gives an odd answer, causing rejection later.
34.		Scan across under the letter. If something is there, reject.
35.		Up 1 unit should be in the letter.
36.		Count three bars up. If not, reject.
38.		Back down (to the upper loop of a "B").
39.		Check "E" or "B" by looking for a right-hand bar.

42.	EGREG2	An alternative hypothesis testing function.
43.		The picture is again hit on its left hand edge.
44.		Move one unit into the letter.
45.		Move up to the top; step back in.
46.		Move to the bottom; step back in.
47.		Find the height.
48.		Move right height 1/5 (inside the bar).
49.		Move up two bars. If not two bars FAIL.
50.		Move back into the middle bar; keep note of X position.
51.		Move left, and if outside the original line, fail.
52 - 54.		Move right to the end, then back to middle.
55.		Move up two bars. If not two bars, fail.
56.		Fail if the top is not flat, if 57 is higher than 45.
57.		Move back into the bar.
58.		Move right to the end.
59.		Move back to the middle and 1/5 height down.
60.		Test for right-hand bar of "B".
70 - 83.	GREGTEST	Applied EGREG1 or EGREG2 to the whole alphabet, and prints the results.

[GREGORY]   [23.26 27 5 75]

```
FUNCTION MOVEFN BW XINC YINC XHI YHI;
VARS B;
IF BFN(XNOW,YNOW)=BW THEN ELSE 1; EXIT;
 LP:
MOVE(XNOW+XINC,YNOW+YINC);
IF (XHI-XNOW)*XINC<0 THEN 1; EXIT;
IF (YHI-YNOW)*YINC<0 THEN 1; EXIT;
BFN(XNOW,YNOW)->B;
IF B=BW THEN ELSE 0; EXIT;
GOTO LP; END;

VARS MOVEUP MOVEDOWN MOVERIGHT MOVELEFT;
MOVEFN(%1,0,100,100%)->MOVERIGHT;
MOVEFN(%0,1,100,100%)->MOVEUP;
MOVEFN(%0,-1,0,0%)->MOVEDOWN;
MOVEFN(%-1,0,0,0%)->MOVELEFT;

FUNCTION STARTSCAN;
LP: MOVE(0,YNOW);
IF 0.MOVERIGHT THEN YNOW-7->YNOW;

   IF YNOW<0 THEN 100+YNOW->YNOW; CLOSE; GOTO LP; CLOSE;
END;


FUNCTION EGREG1;
.STARTSCAN;
MOVE(XNOW+1,YNOW);
1.MOVEUP.ERASE; MOVE(XNOW,YNOW-1);
YNOW->YMAX;
1.MOVEDOWN.ERASE; YNOW->YMIN; YMAX-YMIN->HEIGHT;
IF MOVEFN(0,1,0,XNOW+(HEIGHT*(1/5)),0) THEN ELSE [1];  EXIT;
MOVE(XNOW,YNOW+1);
IF 1.MOVEUP OR 0.MOVEUP OR 1.MOVEUP OR 0.MOVEUP OR 1.MOVEUP
   THEN [2]; EXIT;
MOVE(XNOW,YNOW-(HEIGHT*(2/10)));
IF MOVERIGHT(0) THEN [E] ELSE [B]; CLOSE;
END;

FUNCTION EGREG2;
.STARTSCAN;
MOVE(XNOW+1,YNOW);
1.MOVEUP.ERASE; MOVE(XNOW,YNOW-1);
1.MOVEDOWN.ERASE; MOVE(XNOW,YNOW+1);
.LIMITS;
MOVE(XNOW+(HEIGHT*(1/5)),YNOW);
IF 1.MOVEUP OR 0.MOVEUP OR 1.MOVEUP THEN [1]; EXIT;
MOVE(XNOW,YNOW-1); XNOW->XK;
1.MOVELEFT.ERASE; IF XNOW+1<XMIN THEN [2];EXIT;
MOVE(XNOW+1,YNOW);
IF 1.MOVERIGHT THEN [3]; EXIT;
MOVE(XK,YNOW);
IF 1.MOVEUP OR 0.MOVEUP OR 1.MOVEUP THEN [4]; EXIT;
IF YNOW-1>YMAX THEN [5]; EXIT;
MOVE(XNOW,YNOW-1);
```

```
IF 1.MOVERIGHT THEN [6]; EXIT;
MOVE(XK,YNOW-(HEIGHT*(2/10)));
IF 0.MOVERIGHT THEN [E] ELSE [B]; CLOSE;
END;


VARS TESTSON GREGFN;
1->TESTSON; EGREG1->GREGFN;

FUNCTION LINPR L;
APPLIST(L,LAMBDA S; 1.NL; S.PR; END;);END;

FUNCTION GREGTEST XLO YLO SIZE;
VARS ALPH RESULT;
33->ALPH;
LP:
LETTER(CONSWORD(ALPH,1),XLO,YLO,SIZE)->BFN;
CUCHAROUT; CHAROUT->CUCHAROUT;
ALPH.CHAROUT; 2.SP;.RESET;
.GREGFN; ->RESULT; RESULT.PR; ->CUCHAROUT;
[% ALPH,1.CONSWORD.RESULT,XLO,YLO,SIZE,[EGREGORY],DISTANCE,
DISTB,CROSSINGS%].PR; 1.NL;
IF TESTSON THEN .DEPOSE;POINTLIST.REV.LINPR;
CLOSE;
ALPH+1->ALPH; IF ALPH<59 THEN GOTO LP; CLOSE;
END;
```

5.	STENCIL	Recognising function.
8.	LIMITS	Finds the position, height and width.
9, 10.		Go through the alphabet, setting up each letter to test against the delineation.
12, 13.		Count the points in the delineation POINTLIST, outside the test letter.
14, 15.		If the errors are less than a small fraction of the total points, add this letter to result.
16, 17.		Print what has been done.
23.	STENTEST	Supervisory function.
26.		Set up a letter.
27.		Type out that letter.
28.		Scan it, try to recognise it.
29, 30.		If distance is not too great & nothing has been recognised go back & scan some more, otherwise print the result.
31.		Go on to next letter.

[STENCIL]   [24.53 28 4 75]

```

VARS ERRORS ERRFRACT RESULT TESTSON;
1->TESTSON;  5/100->ERRFRACT;

FUNCTION STENCIL;
VARS TESTFN;
[]->RESULT;
.LIMITS;
APPLIST([ A B C D E F G H I J K L M N O P Q R S T U V W X Y Z],
LAMBDA S; LETTER(S,XMIN,YMIN,HEIGHT)->TESTFN;
0->ERRORS;
APPLIST(POINTLIST,LAMBDA Q;
 IF TESTFN(Q.HD,Q.TL.HD) THEN ELSE ERRORS+1->ERRORS; CLOSE;END;);
IF ERRORS/POINTLIST.LENGTH<ERRFRACT
   THEN S::RESULT->RESULT; CLOSE;
 IF TESTSON THEN 1.NL; APPLIST([% S,ERRORS,POINTLIST.LENGTH,
XMIN,YMIN,HEIGHT%],LAMBDA S; S.PR; END;); CLOSE;
END;);
1.NL; RESULT.PR; END;



FUNCTION STENTEST XLO YLO SIZE SCAN;
VARS A;
33->A;
LP: LETTER(CONSWORD(A,1),XLO,YLO,SIZE)->BFN;
A.CUCHAROUT; 1.NL;.RESET;
L2: .SCAN;.STENCIL;
IF DISTANCE<10000 THEN ELSE
 IF RESULT.NULL THEN GOTO L2;  CLOSE; CLOSE;
RESULT.PR;
A+1->A; IF A<60 THEN  GOTO LP; CLOSE;
END;

```

JALL

The data comes from the experiment as a paper tape with characters X, Y, Z repeated until the end of trial is signalled by a length of blank tape. X & Y are 8-bit position coordinates, Z is 0 (white) or 16 (black). This first pass through the data calculates statistics, applies some tests, and prints a drawing of the trial.

[JA_L]   [12.43 28 JUNE 1971]

```

VARS L1 GENFN;

FUNCTION GO;
APPLIST(L1,GENFN);
END;

DCOMP->GENFN;
[[JREAD][JHIST][JBUFFER][JUNKSHOW][COLUMNS][JTEST]
[JRAND][JDENMAT][JRUN]]->L1;
.GO;
```

JRAND

If a string of points are sample of a random walk taken at equal time intervals, then the sum of distances between points = 2 x the sum of distances between alternate points. This set attempted to assess proximity to random walk for the scan.

(S2OLX(N), S2OLY(N)) is the previous point with a sequence number divisible by 2^N. ST2ODIST(N) is the sum of distances between such points.

4.	RANDSET	sets this up.
17.	RWALKQ	loads each point.
28.	PRAND	prints the result.

[JRAND]   [12.43 28 JUNE 1971]

```
VARS S20LX S20LY ST20DIST ;

FUNCTION RANDSET;
SUBSCR(%INIT(20)%)->S20LX;
SUBSCR(%INIT(20)%)->S20LY;
SUBSCR(%INIT(20)%)->ST20DIST;
VARS N; 1->N;
LP:
128->S20LX(N); 128->S20LY(N); 0->ST20DIST(N);
IF N<20 THEN N+1->N; GOTO LP; CLOSE;
END

FUNCTION DISTOF A B C D ;
SQRT((A-C)*(A-C)+(B-D)*(B-D)); END;

FUNCTION RWALKO;
VARS POWER EXP2; 1->POWER; 1->EXP2;
LO:
IF (TIME//EXP2;.ERASE;) =0 THEN
DISTOF(O1X,O1Y,S20LX(POWER),S20LY(POWER))
 +ST20DIST(POWER)->ST20DIST(POWER);
 O1X->S20LX(POWER); O1Y->S20LY(POWER);
POWER+1->POWER; EXP2*2->EXP2;
 GOTO LO; CLOSE;
END;

FUNCTION PRAND;
VARS X;
3.NL; 'RANDOM WALK TEST'.PR;
FOR X, STEP(2,1,20) DO
IF ST20DIST(X)=0 THEN EXIT;
1.NL; 'D'.PR;(X-2).PR;'/'.PR; (X-1).PR; 3.SP;
 ST20DIST(X-1)/ST20DIST(X):3,3.PRINTRL; REPEAT;
END
```

JDENMAT

DENMAT is a matrix of densities, to find which areas are most densely scanned.

8.	SETDENMAT	sets up the matrix.
12.	DENSET	loads it.
20.	PRDENMAT	prints it.

```
[JDENMAT]     [ 12.42 28 JUNE 1971]


VARS DENMAT XBASE YBASE XIO YIO MATCOARSE
 MATHEIGHT MATWIDTH;

15->XBASE; 15->YBASE; 15->MATCOARSE;
11->MATHEIGHT; 11->MATWIDTH;

FUNCTION SETDENMAT;
NEWARRAY([%0,MATWIDTH,0,MATHEIGHT%],LAMBDA I J ;0;END;);
->DENMAT; END;

FUNCTION DENSET;
INTOF((Q1X-XBASE)/MATCOARSE)->XIO;
IF XIO<0 OR XIO>MATWIDTH THEN EXIT;
INTOF((Q1Y-YBASE)/MATCOARSE)->YIO;
IF YIO<0 OR YIO>MATHEIGHT THEN EXIT;
1+DENMAT(XIO,YIO)->DENMAT(XIO,YIO);
END;

FUNCTION PRDENMAT;
VARS PT X Y ;
PRINTRL(%5,0%)->PT;
3.NL; 'DENSITY MATRIX'.PR; 2.NL;
7.SP;
FOR X, STEP(0,1,MATWIDTH) DO
 XBASE+X*MATCOARSE;.PT; REPEAT; 2.NL;
FOR Y, STEP(0,1,MATHEIGHT) DO
 YBASE+Y*MATCOARSE;.PT; 1.SP;
 FOR X, STEP(0,1,MATWIDTH) DO
  DENMAT(X,Y).PT;REPEAT; 2.NL; REPEAT;
END;
```

JUNKSHOW

A page is held as an array of points, 69 x 69, and is a display of the scan.

9.	DISPLAY	prints a general array.
41.	DISPAGE	prints this one.
43.	LPAGE	loads a new point into the page.

```
[JUNKSHOW]    [ 12.43 28 JUNE 1971]


VARS FUNCTION ( INIT4 NEWYARRAY    );
VARS PSPECN PSPEC4  PAGE ;
'  .%*' ->PSPEC4;
''ABCDEFGHIJKLMNOPQRSTUVWXYZ[$]↑''->PSPECN;
63->SUBSCRC(32,PSPECN);
32->SUBSCRC(1,PSPECN);

FUNCTION DISPLAY FN PSPEC INC BLIST;
VARS Q X, Y X0 X1 Y0 Y1;
BLIST.NEXT.NEXT.NEXT.NEXT.ERASE; ->Y1->Y0->X1->X0;
FUNCTION PRAX;
1.NL;1.SP; X0->X;
LF: IF INC>2 THEN 1.SP CLOSE;
     X.INTOF//10; .ERASE; .CUCHAROUT;X+INC->X;
 IF X=<X1 THEN GOTO LF CLOSE;
END;

IF PSPEC=UNDEF THEN PSPECN->PSPEC CLOSE;
NL(2);
[%X0,Y1%].PR; 45.SP; [%X1,Y1%].PR; .PRAX;
  Y1->Y;
LY: X0->X; NL(1); Y.INTOF//10; .ERASE; .CUCHAROUT;
  IF INC>2 THEN 1.SP CLOSE;
LX:

SUBSCRC(FN(X.INTOF,Y.INTOF)+1,PSPEC);
   .CUCHAROUT;
  IF INC>2 THEN 1.SP CLOSE;
  INC+X->X; IF X=<X1 THEN GOTO LX CLOSE;
  Y.INTOF//10; .ERASE; .CUCHAROUT;
  Y-INC->Y; IF Y>=Y0 THEN GOTO LY CLOSE;

 .PRAX;1.NL;
[%X0,Y0%].PR; 45.SP; [%X1,Y0%].PR;
 3.NL;
END;

VARS FUNCTION(,SUB2,INIT2,DISPAGE );
STRIPFNS("PIC",0,2)->SUB2->INIT2;
DISPLAY(%,PSPEC4,1,[6 69 6 69], %)->DISPAGE;

FUNCTION LPAGE;
VARS X Y Z;
01.NEXT.NEXT.ERASE;INTOF(/3)->Y; INTOF(/3)->X;
IF X<6 OR Y<6 OR X>69 OR Y>69 THEN EXIT;
PAGE(X,Y)->Z;
IF Z=(BW0+1) OR Z=3 THEN EXIT;
IF Z=0 THEN (BW0+1) ELSE 3 ;CLOSE; ->PAGE(X,Y);
END;
```

JTEST

NOTEST	is true for normal running, false for testing, when the program prints intermediate values as it runs.
13. HEADER	prints headings.

[JTEST]   [12.43 28 JUNE 1971]

```
VARS NOTEST;
FUNCTION TEST;
IF NOTEST THEN EXIT;
COL(5,Q1);COL(15,Q2);COL(25,Q3);
COL(35,ANGLE);COL(42,CROSS);COL(47,", ");
PRINTRL((DIST+(1000*DISTK)),6,0);2.SP;
PRINTRL((DISTB+(1000*DISTBK)),6,0);
2.SP; RW0.PR; 2.SP; (TIME/10).PR;1.NL; END;



FUNCTION HEADER;
COL(5,"Q1"); COL(15,"Q2"); COL(25,"Q3");
COL(35,"ANGLE"); COL(42,"CROSS"); COL(49,"DISTANCE");
COL(60,"WRITEDIST");COL(66,"RW"); COL(71,"TIME");
END;
```

JHIST

11.	HISSHOW		prints out a histogram.
29.	ZERO		loads a histogram with zeros.
35.	TRANSHIS		transfers a histogram.
39.	LHIST		loads histograms.
		HISTANG	is of direction of movement.
		HISDIST	is of rate of movement.
51.	WASH		resets a histogram.

[JHIST]   [12.42 28 JUNE 1971]

```
VARS MARKCHAR Z LENGTH1 HYOX ARRAY
FUNCTION ( PRO HISTANG HISTANG1 HISDIST HISDIST1 );
26->MARKCHAR;

FUNCTION RUNTHRO F;
0->Z;
IL: Z.ARRAY.F;Z+1->Z;IF Z=< LENGTH1 THEN GOTO IL; CLOSE;
END;

FUNCTION HISSHOW ARRAY LENGTH1 SCALE;
VARS M; -1000->M;
PRINTRL(% 3,0 %)->PRO;
2.NL;
RUNTHRO(LAMBDA S; S/SCALE->Z.ARRAY;END;);
RUNTHRO(LAMBDA S; IF S>M THEN S->M; CLOSE; END;);
IF M>15 THEN  15/M ELSE 1 ; CLOSE; ->OX;
RUNTHRO( LAMBDA S; S*OX->Z.ARRAY;END;); 15->M;
M.INTOF->M; JL:
1.NL;( M/OX).PRO;
RUNTHRO(LAMBDA S; IF S>M THEN MARKCHAR ELSE 16;CLOSE;
.CUCHAROUT;1.SP; END;);
IF M THEN M-1->M; GOTO JL; CLOSE;NL(1);4.SP;
RUNTHRO(LAMBDA S; Z//10;.ERASE;.CUCHAROUT;1.SP;END;);1.NL;4.SP;
RUNTHRO(LAMBDA T; Z//10;->M; IF =0 THEN M.CUCHAROUT ELSE 1.SP;
CLOSE; 1.SP; END;);
END;

FUNCTION ZERO I ; 0; END;
NEWARRAY([0 035],ZERO)->HISTANG;
NEWARRAY([0 25],ZERO)->HISDIST;
NEWARRAY([0 25],ZERO)->HISTANG1;
NEWARRAY([0 25],ZERO)->HISDIST1;

FUNCTION TRANSHIS ARRAY AR2 LENGTH1;
RUNTHRO( LAMBDA S; Z.AR2+S->Z.AR2;0->Z.ARRAY;END;);
END;

FUNCTION LHIST;
INTOF(ANGLE*4)->HY;
HISTANG(HY)+1->HY.HISTANG;
INTOF(DD)->HY;
IF HY>25 THEN EXIT;
HY.HISDIST +1->HY.HISDIST;
TIME//100; .ERASE; IF =0 THEN
TRANSHIST(HISTANG,HISTANG1,25);
TRANSHIS(HISDIST,HISDIST1,25);
CLOSE;
END;

FUNCTION WASH ARRAY;
RUNTHRO( LAMBDA S; 0->Z.ARRAY; END;);
END;
```

JREAD

1.	DISTK		POP-2 numbers are only 14 bits, and thus scanning thousands of small numbers is unreliable. Here distance is represented by DIST, (< 1000) and DISTK, thousands of units.
8.	BLANK		TRUE if a character is either 0 (BW = 0) or 16 (BW = 1). These are the only values the brightness can have.
12.	START		reads in blank type.
		TIN	reads one character from the buffer (in JBUFFER).
20.	READ3		reads in one point.
		Q4X	stores the X value of the fourth point back.
		BW	is the black/white value at the point.
		ANGLE	is the direction of scan now.
		DIST	is distance.
		DISTB	is writing distance.
		CROSS	is crossings.

```
[JREAD]    [ 12.41 28 JUNE 1971]

VARS DIST DISTK TIME CROSS ANGLE BW0 BW1 DISTB DISTBK N PI RUN  Q
 A RUNEND Q1 Q2 Q3 DX DY DD
QX Q1X Q2X Q3X Q4X Q5X QY Q1Y Q2Y Q3Y Q4Y Q5Y TIN; ;
128->Q1X:128->Q1Y;
[128 128]->Q1; 0->N; 0->BW1; 3.14159 ->PI; [128 128]->Q;0->U;
128->Q2X: 128->Q2Y: 128->Q3X: 128->Q3Y; 128->Q4X: 128->Q4Y;

FUNCTION BLANK;
 ->A:
IF A=0 OR A=16 THEN TRUE ELSE FALSE; CLOSE: END;

FUNCTION START;
L1:
IF .TIN.BLANK THEN N+1->N; GOTO L1; CLOSE;
L2:
IF .TIN.BLANK .NOT THEN GOTO L2; CLOSE: END;

[INVTRIG].LIBRARY.COMPILE;

FUNCTION READ3:
VARS ;0->N;L0:
.TIN->QX: .TIN->QY:L17: .TIN->U;
IF U.BLANK.NOT  THEN  N+1->N: GOTO L17; CLOSE;
IF N THEN 'SCRAMBLED, BLANKS='.PR;N.PR; 1.NL:CLOSE:
IF QX.BLANK AND QY.BLANK THEN IF RUN   THEN   0->RUN:GOTO L0: CLOS
RETURN ELSE 1->RUN:CLOSE;
Q4X->Q5X:Q3X->Q4X;Q2X->Q3X: Q1X->Q2X: QX->Q1X;
Q4Y->Q5Y:Q3Y->Q4Y;Q2Y->Q3Y: Q1Y->Q2Y: QY->Q1Y; BW0->BW1;
IF U=0 THEN 0 ELSE 1: CLOSE; ->BW0;
(Q1X - Q2X)->DX; (Q1Y - Q2Y)->DY;
IF DX=0 THEN PI/2 ELSE ARCTAN(DY/DX):CLOSE->ANGLE;
IF ANGLE<0 THEN ANGLE + PI ->ANGLE; CLOSE;
IF ANGLE=0 AND DX<0 THEN ANGLE+PI->ANGLE: CLOSE;
IF DY<0 THEN ANGLE+PI->ANGLE: CLOSE;
SQRT((DX*DX)+(DY*DY))->DD;
DD+DIST->DIST;
IF DIST>1000 THEN DISTK+1->DISTK: DIST-1000->DIST: CLOSE:
IF BW0 THEN DD+DISTB->DISTB; CLOSE;
Q1X->Q1.HD: Q1Y->Q1.TL.HD;
IF DISTB>1000 THEN DISTBK+1->DISTBK; DISTB-1000->DISTB: CLOSE;
TIME+1->TIME;
IF BW0=BW1 THEN ELSE CROSS+1->CROSS; CLOSE;
END;
```

JBUFFER

The paper tape reader is not well adapted to reading 3 characters and stopping many times. This reads in characters in batches of 300.

7.	BLOAD	reads in 300 characters.
15.	TIN	produces characters one by one for other processes, reloading the buffer when necessary.

COLUMNS

8. Character 17 is new line.

12. COL (N, Z) prints Z starting at column N.

[COLUMNS]   [12.4 28 JUNE 1971]

```
VARS LPCOL; 1->LPCOL;
CHAROUT->COW;
'POPMESS FOR LP80 ->COW'.PR; 1.NL;
'AFTER ERROR, PIG->CUCHAROUT'.PR;

FUNCTION PIG;
VARS X; LPCOL+1->LPCOL; ->X;
IF X=17 THEN 1->LPCOL;CLOSE; X.COW; END;

PIG->CUCHAROUT;

FUNCTION COL N Z;
VARS Q;
L:
(N-LPCOL)->Q; IF Q<0 THEN 1.NL;GOTO L ; CLOSE;
Q.SP;IF Z=0 THEN 1.SP ELSE Z.PR;CLOSE; END;
```

JRUN

5.	MAMMON	Processes one trial.
6 - 12.		Set up initial conditions.
15.		Run through the tape.
18.	MICHAEL	Processes one trial. Read in 3 characters (X, Y, Brightness) (In JREAD) Load into a page (In JUNKSHOW) Print them out if testing (NOTEST = 0). (In JTEST) Load histograms (In JHIST) Test for random walk (In JRAND) Load density matrix (In JDENMAT)
23.	DATAOUT	Print out results.
57.	JRUN	Carry on to next trial.

```

VARS W NAME RLPTOUT
FUNCTION ( MICHAEL DATAOUT);

FUNCTION MAMMON;
.INITCOND;
0->DIST; 0->DISTK; 0->DISTR; 0->DISTRK:
0->CROSS: 0->ANGLE: 0->TIME;1->RUN:PIG->CUCHAROUT;
NEWANYARRAY([6 69 6 69],LAMBDA A B: 0; END:,INIT2,SUR2)->PAGE;
FUNCTION F2:.TREAD:->W; W.PR: W; END:
.RANDSET: .SETDENMAT:
0->N:
.START:
IF NOTEST THEN ELSE .HEADER;F2 -> TIN;CLOSE;
.MICHAEL: END;
FUNCTION MICHAEL:
L0:
.READ3: .LPAGE: .TEST; .LHIST;.RWALK0; .DENSET:
IF RUN THEN GOTO L0; CLOSE:
.DATAOUT:
END;

FUNCTION DATAOUT:
IF TIME <8 THEN EXIT:
64.CUCHAROUT:
PAGE.DISPAGE;
3.NL: .GEOMOUT;
2.NL:
COL(10,'DISTANCE'): COL(25,DISTK);",".PR: DIST.PR;
COL(10,'TIME'):COL(25,(TIME/10));1.SP; 'SECONDS'.PR;
COL(10,'CROSSINGS');COL(25,CROSS);
COL(10,'WRITING DISTANCE'):COL(35,DISTRK);",".PR: DISTR.PR:
COL(10,'DISTANCE PER CROSSING');
  COL(35,((DIST+(1000*DISTK))/CROSS));
COL(10,'AVERAGE VELOCITY');COL(35,(((DIST+(1000*DISTK))*10)/TIME)
1.SP;'UNITS PER SECOND'.PR:

RLPTOUT->CUCHAROUT:
NAME.PR; 1.NL;
DISTK.PR:DIST.PR:1.NL;
DISTRK.PR;DISTR.PR:1.NL;
TIME.PR; 1.NL;
CROSS.PR: 1.NL:
PIG->CUCHAROUT:
 64.CUCHAROUT;HISSHOW(HISTANG1,25,5):
1.NL: 'ANGLES*4'.PR;
HISSHOW(HISDIST1,25,10);
1.NL: 'DISTANCE'.PR;
25->LENGTH1;
HISTANG.WASH:
HISDIST.WASH:
HISTANG1.WASH;
HISDIST1.WASH;
3.NL;
.PRAND: .PRDENMAT;
END;
 FUNCTION JRUN:




 LP: .MAMMON;  GOTO LP: END;
```

JCELLALL

This is a second pass through the data to find the data for the cells surrounding the letter. The pictures produced by the first pass are checked, and a tape prepared, giving the subject, the scan type, the transparency (and the target), and deviations from the normal position of the letters.

This is processed by this pass to give the cell data.

[JCELLALL]   [12.4 28 JUNE 1971]

```
VARS JL GENFN;

FUNCTION GO;
APPLIST(JL, LAMBDA S: [% S%].GENFN; END;); END;

[FOR].LIBRARY.COMPILE;

[ JREAD JGEOM JBUFFER1 JHIST COLUMNS GRAPHOF JCELLS1 JCELLS2
JCELLS3 JCELLS4 JCELLS5 JCELLS6 JCELLS7]->JL;
DCOMP->GENFN;
```

JBUFFER1

Again a buffer from the tape punch is required.

9.	BBLOAD	loads a buffer.
15.	TREAD	reads out of it.
22.	RUNLOAD	loads a list of buffers with a whole trial, this gives the length of the trial (TIMEST) to be checked against the other data.
35.	TIN	reads out of the list of buffers (LISNOW)
46.	CHECKNAME	loads a trial, checks its length, processes it, and repeats. (RECL holds the list of other data.)

[JBUFFER1]   [12.41 28 JUNE 1971]

```
'PTIN->TPIN'.PR;
VARS LISNOW NOWON SUBL8 INITL8 BLNOW BLNUMB LISBUFF TIMEST
BPOS BB300 TPIN;
STRIPFNS("BUFFER1",20,8)->SUBL8->INITL8;
0->LISNOW; []->LISBUFF;
INITL8(300)->BB300; 0->BPOS;

FUNCTION BBLOAD;
VARS N; 300->N;
LB: .TPIN->SUBL8(N,BB300); N-1->N;
 IF N THEN GOTO LB; CLOSE;
300->BPOS; END;

FUNCTION TREAD;
LT: IF BPOS THEN SUBL8(BPOS,BB300); BPOS-1->BPOS;
 ELSE .BBLOAD; GOTO LT; CLOSE;
END



FUNCTION BUNLOAD;
VARS CURSTR AA A;
[]->LISBUFF; 0->BLNUMB;
L7: IF .TREAD.BLANK THEN GOTO L7; CLOSE;
LS: INITL8(300)->CURSTR;300->NOWON;
LP: .TREAD->A; IF A.BLANK THEN 1+BLNUMB->BLNUMB; ELSE 0->BLNUMB;C
 IF BLNUMB>6 THEN REV(CONS(CURSTR,LISBUFF))->LISBUFF;
   LISBUFF.LENGTH*100-NOWON/3;->TIMEST; EXIT;
A->SUBL8(NOWON,CURSTR); NOWON-1->NOWON;
IF NOWON THEN GOTO LP; CLOSE;
CURSTR::LISBUFF->LISBUFF;GOTO LS;
END;

FUNCTION TIN;
IF LISNOW THEN SUBL8(LISNOW,CURSTR);LISNOW-1->LISNOW;
  ELSE IF LISBUFF.NULL THEN .BUNLOAD; CLOSE;
   300->LISNOW; LISBUFF.NEXT->LISBUFF->CURSTR;
.TIN; CLOSE;
END;


VARS SAVEL DUML RLIST TIMESTER TIMESTVAR;
[]->SAVEL; 0->TIMESTER;10->TIMESTVA

FUNCTION CHECKNAME;
VARS AA TNAMME TNAME TTARGET;
L7:.TIN;0->LISNOW; .BUNLOAD;
 IF ABS(RECL.HD.RTIME-TIMEST+TIMESTER)<TIMESTVAR
     THEN  RECL.HD.GARRIEL;
      RECL.HD::SAVEL->SAVEL;
       RECL.TL->RECL;
       GOTO L7;
     ELSE CHAROUT->CUCHAROUT; 1.NL;
       RECL.HD.DATALIST.PR; 1.NL;
       TIMEST.PR;1.NL;.POPREADY; GOTO LZ;
    CLOSE;
```

8.	BUZZ	takes the next item from a list, even a negative number.
13.	RIN	makes a record of the data and puts it into a list (RECL).
25.	RECLEDIT	Edits RECL.
31.	RECLREAD	reads in RECL.

[JCELLS1]   [12.42, 28 JUNE 1971]

```
COMMENT TO READ IN THE TAPE TO A RECORD;
VARS RECL RNAME  RTIME RCROSS RTARGET RDIST RDISTB RSCAN CONSD4 DE
KEPTL RECLEDIT RFRAME   RXSHIFT RYSHIFT RSUBSHIFT;

[]->RECL;

FUNCTION BUZZ L;
L.HD->ZO:
IF ZO="-" THEN L.TL->L: (L.HD*(-1)); ELSE ZO: CLOSE;
L.TL: END;

FUNCTION RIN:
[]->RECL:
RECORDENS("DATA4",150,[ 0 0 0 0 0 0 0 0 0 0 0])
->RSUBSHIFT->RYSHIFT->RXSHIFT->RSCAN->RTARGET->RFRAME->RCROSS->
RTIME->RDISTB->RDIST->RNAME->DESTD4->CONSD4:
LO:
.LISTREAD.BUZZ.BUZZ.BUZZ.BUZZ.BUZZ.BUZZ.BUZZ.BUZZ.BUZZ.BUZZ.BUZZ.B
.CONSD4::RECL->RECL;
GOTO LO;
 .RECLEDIT: RECL->KEPTL;
END;

FUNCTION RECLEDIT;
APPLIST(RECL, LAMBDA S; IF EQUAL(S.RNAME,[PETER MITCHELL]) AND
   S.RTARGET="ERCK" THEN 5->S.RXSHIFT ; 3->S.RYSHIFT;CLOSE: END;);
END;

[JSTATS8].DCOMP;
FUNCTION RECLREAD TNAME;
[PTIN]<>TNAME;.POPMESS->DIN;
DIN.INCHARITEM.FNTOLIST->PROGLIST;
.RIN: RECL.REV->RECL:.VARTARE;
END;

'RECLREAD(<TNAME>)'.PR;
```

JSTATS8

The incoming data has a word "EGN". This converts it to a strip 'E G N', which corresponds to the letter positions.

51. VARTARE converts word to strip.

```
[JSTATS8]    [ 12.4 28 JUNE 1971]



[FOR].LIBRARY.COMPILE;
VARS RECL   PTNAME

B B1 B2 P L G EB EH EI ET EF JUB CBF RBM DBN KBN QBEH QHEB
VIAB JSPB BSSI VBUMA NPBUO SONZB MDCEB NDBWD W Q K BO BP BQ BU BW
J EGN ZEA MXE ONCE ROSE MEVR ERCK
E1 E2 E H   CODE JXZEC DEDPH EJMFO KORTE LPCIE ;
[ PTIN SERIES FOUR MK3A]->PTNMAE:PTNMAE->PTNAME;
FUNCTION RECLIN;
[]->RECL;
[]:[ PTIN]<> PTNAME; .POPMESS->DATIN;
DATIN.INCHARITEM.FNTOLIST->PROGLIST;
LQ:
.LISTREAD; FOR X,STEP (1,1,36) DO .NEXT; REPEAT;
.ERASE; .CONCEIL::RECL->RECL;
GOTO LQ;
END;

"E"->E:"B"->B;
' E    '->E1;      '   B '->B1:
'    E '->E2;      '  B   '->B2:

'W     '->W;       ' P    '->P;
'   K  '->K:       '     G'->G;
'     Q'->Q:       'L     '->L;
'  BU  '->BU;      ' B E  '->EB:
'BP    '->BP;      ' EF   '->EF;
' BQ   '->BQ;      ' H E  '->EH:
' WB   '->BW;      ' EI   '->EI;
'O  B  '->BO;      ' E T  '->ET;
'EDC   '->EDC;     'J UB  '->JUB;
'E ZJ  '->EZJ:     ' CBF  '->CBF;
'ZEA   '->ZEA;     'RBM   '->RBM;
'E G N'->EGN;      ' DBN  '->DBN;
' MX E'->MXE:      'K BN  '->KBN;
'M EVR'->MEVR;     'Q BEH'->QBEH:
'ROSE '->ROSE;     'QHE B'->QHEB:
'E RCK'->ERCK;     'VIAB '->VIAB:
'CO OE'->CODE;     'JSP B'->JSPB;
'ONCE '->ONCE;     ' BSSI'->BSSI:
'KORTE'->KORTE:    'VBUMA'->VBUMA;
'LPCIE'->LPCIE:    'NPBUO'->NPBUO;
'EJMFO'->EJMFO:    'SONZB'->SONZB;
'DEDPH'->DEDPH:    'MCDEB'->MCDEB;
'JXZEC'->JXZEC:    'NDBWD'->NDBWD;

VARS UB OB WB QB BE FE HE TE IE;
BU->UB;BP->PB;BQ->QB;BO->OB;BW->WB;
EB->BE:EF->FE;EH->HE;EI->IE;ET->TE;
FUNCTION VARTARE:
APPLIST(RECL, LAMBDA S: S.TARGET.VALOF->H;
IF H=E THEN IF S.DB2>S.DB4 THEN E1 ELSE E2:CLOSE:ELSE H; CLOSE:->S
T;
IF H=B THEN IF S.DB4>S.DB3 THEN B1 ELSE B2 CLOSE: ->S.TARGET; CLOS
END;); END:
```

6. LINEUP

sets the cell boundaries at their normal positions plus any corrections read in.

RXSHIFT & RYSHIFT move the whole matrix.

RSUBSHIFT contains a list of instructions to move individual letter boundaries. This was usually necessary for "W" & "M".

```
[JCELLS2]     [ 12.41 28 JUNE 1971]

VARS XD1 XD2 XD3 XD4 XD5 XU1 XU2 XU3 XU4 XU5
YD1 YD2 YD YD3 YD4 YD5 YU1 YU2 YU3 YU4 YU5
:

FUNCTION LINEUP;
VARS Q Z1 Z2:
25+S.RXSHIFT->Q;
Q->XU1:Q->XU4;Q->XD3:
36+S.RXSHIFT->Q; Q->XU3;Q->XD2:Q->XD5;
47+S.RXSHIFT->Q;  Q->XU2: Q->XU5;
13+S.RXSHIFT->Q;   Q->XD1;Q->XD4;
16+S.RYSHIFT->Q;  Q->YD4:Q->YD5;
26+S.RYSHIFT->YD3;
34+S.RYSHIFT->Q;  Q->YU4:Q->YU5;Q->YD1:Q->YD2;
42+S.RYSHIFT->YU3;
52+S.RYSHIFT->Q;  Q->YU1:Q->YU2;
S.RSUBSHIFT->Q:
L8:
IF Q.NULL THEN EXIT;
Q.BUZZ.BUZZ->Q->Z2->Z1:
Z1.VALOF+Z2->Z1; GOTO L8;
END;
```

JCELLS3

2.	WITHIN	returns true if ZX, ZY is within a rectangle defined by X1, X2, Y1, Y2.
6.	CELLS	are rectangles surrounding each letter position.
10.	INITCELL	sets up a data structure for the cells.

[JCELLS3]   [12.41 28 JUNE 1971]

```
FUNCTION WITHIN ZX ZY X1 X2 Y1 Y2 ;
IF ZX>X1 AND ZX=<X2 AND ZY>Y1 AND ZY=< Y2 THEN TRUE ELSE FALSE;
CLOSE END;

VARS CELLS CONC DESTC CELL IND INDR N INC INT INFN;
5.INIT->CELLS;SUBSCR(% CELLS %)->CELL;RECORDFNS("SUBCELLS",5,[0 0
])
->INFN->INC->INT ->INDR->IND->DESTC->CONC;

FUNCTION INITCELL;
CONC(0,0,0,0,WITHIN(%XD1*3,XU1*3,YD1*3,YU1*3%))->1.CELL;
CONC(0,0,0,0,WITHIN(%XD2*3,XU2*3,YD2*3,YU2*3%))->2.CELL;
CONC(0,0,0,0,WITHIN(%XD3*3,XU3*3,YD3*3,YU3*3%))->3.CELL;
CONC(0,0,0,0,WITHIN(%XD4*3,XU4*3,YD4*3,YU4*3%))->4.CELL;
CONC(0,0,0,0,WITHIN(%XD5*3,XU5*3,YD5*3,YU5*3%))->5.CELL;END;
```

6.	SUBDAT	loads data from each point into the cells. CELLNOW is the present cell. CELLMAT is a matrix of transitions. TRANSLIST is a list of how long the scan stayed in each cell.
31.	CELLTRANS	resets CELLMAT & TRANSLIST.
38.	TRANSPR	prints TRANSLIST.

[JCELLS4]   [12.42 28 JUNE 1971]

```
VARS FNZ  CELLWAS  CELLMAT CELLNOW
CDIST CDISTB CCROSS CTIME TRANSLIST;
0->CELLWAS;

FUNCTION SUBDAT;
VARS N; 5->N;
CELLNOW->CELLWAS;
     LJ: N.CELL.INFN->FNZ; IF FNZ(Q1X,Q1Y) THEN GOTO LL;
           ELSE N-1->N; IF N THEN GOTO LJ; CLOSE CLOSE;
   LL:
  N->CELLNOW;

IF EQ(CELLNOW,CELLWAS).NOT THEN
  CELLMAT(CELLWAS,CELLNOW)+1->CELLMAT(CELLWAS,CELLNOW);
IF (TIME-CTIME)> 3 THEN
 [% CELLWAS,TIME-CTIME,(DIST+1000*DISTK)-CDIST,(DISTB+1000*DISTBK)
   -CDISTB,CROSS-CCROSS %]::TRANSLIST->TRANSLIST;
 TIME->CTIME; (DIST+1000*DISTK)->CDIST; (DISTB+1000*DISTBK)->CDIST
 CROSS->CCROSS; CLOSE;
 EXIT;

IF CELLNOW=0 THEN EXIT;CELLNOW.CELL.IND+DD->CELLNOW.CELL.IND;
IF BW0=1  THEN CELLNOW.CELL.INDB+DD->CELLNOW.CELL.INDB;CLOSE;
CELLNOW.CELL.INT+1->CELLNOW.CELL.INT;
IF BW0=BW1 THEN ELSE CELLNOW.CELL.INC+1->CELLNOW.CELL.INC; CLOSE;
CELLNOW->CELLWAS;
END;


FUNCTION CELLTRANS;
0->CDIST; 0->CDISTB; 0->CCROSS; 0->CTIME; NIL->TRANSLIST;
NEWARRAY([%( -1), 5, ( -1), 5 %], LAMBDA I J ;
  IF I<0 THEN J ; ELSEIF J<0 THEN I; ELSE 0; CLOSE;END;)
->CELLMAT;
END;

FUNCTION TRANSPR;
PIG->CUCHAROUT; 64.CUCHAROUT;
S.RNAME.PR; 6.SP; S.RFRAME.PR; 6.SP; S.RTARGET.PR; 6.SP;
S.RSCAN.PR; 2.NL;
'CELL'.PR;COL(14,'TIME');COL(25,'DIST');COL(35,'DISTB');
COL(46,'CROSS');1.NL;
APPLIST(TRANSLIST, LAMBDA Q;
  Q.HD,PRINTRL(3,0);
  IF Q.HD=0 THEN 5.SP; ELSE 2.SP; SUBSCRC(Q.HD,S.RTARGET.VALOF).CU
T;
  2.SP; CLOSE;
  APPLIST(Q.TL, PRINTRL(%8,1%));
  1.NL;
END;);
END;
```

5.	LHIST1	CTOCNO is the distance between corners.
20.	INITSHOW	LBEGIN is a list of the scan until an edge is found.

[JCELLS5]   [12.41, 28 JUNE 1971]

```
VARS CTOC CTOCNO CRON1 CORN1 ;
0->CORN1:0->CTOC: 0->CTOCNO;

35->LENGTH1;
FUNCTION LHIST1;
CTOCNO+DD->CTOCNO;
IF BW0=BW1 THEN ELSE 1->CTOC; 0->CTOCNO; CLOSE;
IF CORN1=CORNERS THEN  ELSE
  IF CTOC THEN INTOF(CTOCNO/2)->CTOCNO;
     IF CTOCNO>35 THEN  35->CTOCNO; CLOSE;
  CTOCNO.HISTANG+1->CTOCNO.HISTANG; CLOSE;
0->CTOC; 0->CTOCNO:CLOSE:
CORNERS->CORN1:
END;

VARS NOBW LBEGIN LBA;
1->NOBW; []->LBEGIN;

[[35 45][35 160][145 160][145 45]]->LBA;
FUNCTION INITSHOW;
IF NOBW.NOT THEN EXIT;
IF BW0=1 THEN 0->NOBW; EXIT;
[% 01X,01Y %]::LBEGIN->LBEGIN;
END;
```

14.	GABRIEL	sets starting conditions.
21.	MICHAEL	processes one trial.

[JCELLS6]   [12.41 28 JUNE 1971]

```
VARS DISR   SS S ;

FUNCTION GABRIEL S:
CHAROUT->CUCHAROUT:
S.RNAME.PR: 1.NL:
PIG->CUCHAROUT;

.INITCOND;0->CELLWAS:0->CELLNOW; 0->CTOC; 0->CTOCNO; 0->CORN1;  0-
0->CROSS:
0->DIST;0->DISR;0->DISTK:0->DISTRK: 0->DISTB: 0->STILL:0->LINEON:
0->LINENO;0->ANGSUM; 0->CORNERS;
LBA->LBEGIN; 1->NORW: HISTANG.WASH;HISDIST.WASH;
.START;
.LINEUP;
.CELLTRANS;
.INITCELL;
.MICHAEL;
END;

FUNCTION MICHAEL:
L0:
.READ3; .GEOM; .LHIST; .SURDAT:;
IF RUN THEN GOTO L0; CLOSE:
.DATOUT; END;
```

4. DATOUT Prints results.

[JCELLS7]   [12.41 28 JUNE 1971]

```
[FOR].LIBRARY.COMPILE;VARS P5;
PRINTRL(%7,1%)->P5:
FUNCTION DATOUT;
VARS OX FNX;
PIG->CUCHAROUT;
IF TIME<8 THEN EXIT;
S.RTIME/TIME->OX;
IF OX>1.1 OR OX<0.9 THEN CHAROUT->CUCHAROUT;
 S.DATALIST.PR: 1.NL: TIME.PR;
 PIG->CUCHAROUT;1.NL;
' SUSPECTED ERRONEOUS'.PR; CLOSE:
.POPDATE.PR;
COL(1,S.RNAME);COL(25,S.RFRAME);COL(40,S.RTARGET);COL(55,S.RSCAN);
2.NL:
COL(10,'DISTANCE'):COL(25,0);P5(DISTK*1000+DIST):
COL(10,'DISTB');COL(25,0);P5(DISTB+1000*DISTBK);
COL(10,'TIME'):COL(25,0):P5(TIME/10);1.SP; 'SECONDS'.PR;
COL(10,'CROSSINGS');COL(25,0);P5(CROSS);
.GEOMOUT:  3.NL;
COL(40,'SUBCELLS'):
FOR X,STEP(1,1,5) DO COL(22+10*X,X);REPEAT;
FUNCTION AAZ; COL(25,0); FOR X,STEP(1,1,5) DO X.CELL.FNX.P5;
  REPEAT: END;
COL(5,'DISTANCE');IND->FNX: .AAZ:
COL(5,'DISTB'):INDB->FNX;.AAZ;
COL(5,'TIME');INT->FNX: .AAZ;
COL(5,'CROSSINGS'):INC->FNX;.AAZ:
RLPTOUT->CUCHAROUT:
S.DATALIST.PR;
FOR Y,STEP(1,1,5) DO  Y.CELL.DATALIST.PR: 1.NL: REPEAT:
[% ANGSUM,LINELENGTH,LINENO,CORNERS,STILL  %].PR;1.NL:
[%TIME,CROSS,DIST+1000*DISTK,DISTB+1000*DISTBK %].PR; 1.NL:
PIG->CUCHAROUT:3.NL;
'NUMBER OF TRANSITIONS'.PR: 1.NL:
20.SP; 'FROM'.PR: 1.NL;
FOR Y, STEP(-1,1,5) DO 'TO'.PR:
FOR X, STEP(-1,1,5) DO
 PRINTRL(CELLMAT(X,Y),8,0):
 REPEAT: 1.NL; REPEAT;
64.CUCHAROUT:
HISSHOW(HISDIST,25,1);
COL(6,'DISTANCES PER 1/10 SECOND');5.NL;
HISSHOW(HISTANG,25,1);
COL(4,'->');COL(16,"+"); COL(28,'->'); COL(40,1): COL(6,'ANGLES');
.TRANSPR:
64.CUCHAROUT;
END;
```

8. GEOM — calculates various features of the scan.

1. STILL if moved less than 4.5 units in the last 1/2 second.
2. ANGLE is the current direction.
3. CORNER if it doubles back on itself.

21. GEOMOUT — prints this.

[12.42 28 JUNE 1971]

```
VARS PAV5X PAV5Y STILL ANG1 ANG2 DLX DLY ANGSUM A3 A4 A5  P10
LINEON LINELENGTH DANG LINENO CORNERS DVERT DHORIZ:
FUNCTION INITCOND;
128->PAV5X: 128->PAV5Y:0->STILL;0->ANG1; 0->LINEON:0->DHORIZ:
0->LINELENGTH;0->LINENO;  0->CORNERS; 0->ANGSUM;0->DVERT:END:

FUNCTION GEOM;
PAV5X-01X->DLX: PAV5Y-01Y->DLY:
IF DLX*DLX +DLY*DLY <20 THEN STILL+1->STILL; GOTO LT: CLOSE;
ANGLE; ANG2;A3:A4;->A5->A4->A3->ANG2:
ANGLE-A3->DANG:DANG+ANGSUM->ANGSUM;
IF ABS((-2)*01X+02X+3*03X-04X-05X))29 OR
ABS((-2)*01Y+02Y+3*03Y-04Y-05Y)>29 THEN CORNERS+1->CORNERS:CLOSE:
LT:
0.8*PAV5X+0.2*01X->PAV5X: 0.8*PAV5Y+02*01Y->PAV5Y;
DX.ABS+DHORIZ->DHORIZ; DY.ABS+DVERT->DVERT:
END;

PRINTRL(%7,1%)->P10;
FUNCTION GEOMOUT:
COL(10,'TIME STILL'):COL(35,0):P10( STILL/10);
  1.SP:"SECONDS".PR;6.SP: STILL*100/TIME:.PR:'%'.PR;
  ' OF TIME'.PR:
COL(15,'AVERAGE LENGTH');COL(35,0);P10((DIST+1000*DISTK)/IF  CORNERS THE
RS ELSE 1 CLOSE:):
COL(10,'CLOCKWISE MOVEMENT'):COL(35,0);P10(ANGSUM/(2*PI));1.SP: "CIRCLES

COL(10,'CORNERS');COL(35,0);P10(CORNERS):
COL(10,'HORIZONTAL MOVEMENT');COL(35,0);P10(DHORIZ);
  P10(DHORIZ*100/(DIST+1000*DISTK));'% OF DISTANCE'.PR:;
COL(10,'VERTICAL MOVEMENT'); COL(35,0);DVERT.P10;
  P10( DVERT*100/(DIST+1000*DISTK)); '% OF DISTANCE'.PR:
END;
```

00140

GRAPHOF

2.	GRAPHOF (POINTLIST)	prints a display of what has been scanned.
5.		Numbers the points in sequence.
8.		Find the extent of the display.
16.		Set a suitable scale.
23.		64 cucharout sets a new page.
28.		Character 26 is "*". This sets a line of stars.
30.		Start horizontal line.
32.		Catch any points which are on that line.
35.		Start point.
37.		Catch any points at that point.
40.		Print the sequence number of the last point found.

```
DF]     [ 12.4 28 JUNE 1971]


FUNCTION GRAPHOF LIST;
VARS N1 XMIN XMAX YMIN YMAX DIFF INK L1 L2 L3 ;
1->N1;  []->L1;
APPLIST(LIST,LAMBDA X;N1//10:.ERASE:::X->X;N1+1->N1;
      X::L1->L1; END:):  L1->LIST;
-10000->XMAX: -10000->YMAX: 10000->XMIN; 10000->YMIN:
APPLIST(LIST, LAMBDA L:
   IF L.TL.HD<XMIN THEN L.TL.HD->XMIN;CLOSE:
   IF L.TL.HD>XMAX THEN L.TL.HD->XMAX: CLOSE:
   IF L.TL.TL.HD<YMIN THEN L.TL.TL.HD->YMIN; CLOSE:
IF L.TL.TL.HD>YMAX THEN L.TL.TL.HD->YMAX: CLOSE;END;)
:
IF(XMAX-XMIN)>(YMAX-YMIN) THEN (XMAX-XMIN) ELSE (YMAX-YMIN);
CLOSE; ->DIFF;
IF DIFF<60 THEN 1;
ELSEIF DIFF<120 THEN 2;
ELSEIF DIFF<300 THEN 5:
ELSEIF DIFF<600 THEN 10;
ELSE DIFF/50: CLOSE: ->INK;
XMIN-(2*INK)->XMIN: YMIN-(2*INK)->YMIN:
XMIN+(64*INK)->XMAX; YMIN+(64*INK)->YMAX:
64.CUCHAROUT: 2.NL:
XMIN.PR;1.SP;YMAX.PR:45.SP:XMAX.PR;1.SP;YMAX.PR;1.NL:
'INCREMENTS. X='.PR;INK.PR:2.SP; 'Y='.PR: (1.666*INK).PR:1.NL;
64->N1:
LTOP:
26.CUCHAROUT:IF N1 THEN N1-1->N1: GOTO LTOP:CLOSE;
VARS NX NY:38->NY; LIST->L3; 1.NL; 26.CUCHAROUT;
LY:
[]->L1: []->L2: 0->NY;
APPLIST(L3, LAMBDA L: IF L.TL.TL.HD>((NY*INK*1.666)+YMIN) THEN
   L::L1->L1: ELSE L::L2->L2: CLOSE; END:):
L2->L3;
LX:
0->N1;
APPLIST(L1, LAMBDA L: IF L.TL.HD>((NX*INK)+XMIN) THEN
IF L.TL.HD=<(((NX+1)*INK)+XMIN) THEN
 L.HD+1->N1;CLOSE: CLOSE; END:):
IF N1 THEN (N1-1).CUCHAROUT; ELSE 1.SP: CLOSE;
NX+1->NX: IF NX<64 THEN GOTO LX; ELSE 26.CUCHAROUT: CLOSE;
1.NL: 26.CUCHAROUT:
NY-1->NY: IF NY THEN GOTO LY; CLOSE;
64->N1:LR: 26.CUCHAROUT; IF N1 THEN N1-1->N1: GOTO LR; CLOSE:
1.NL;
XMIN.PR;1.SP:YMIN.PR:45.SP;XMAX.PR: 1.SP;YMIN.PR;1.NL;
END;
```